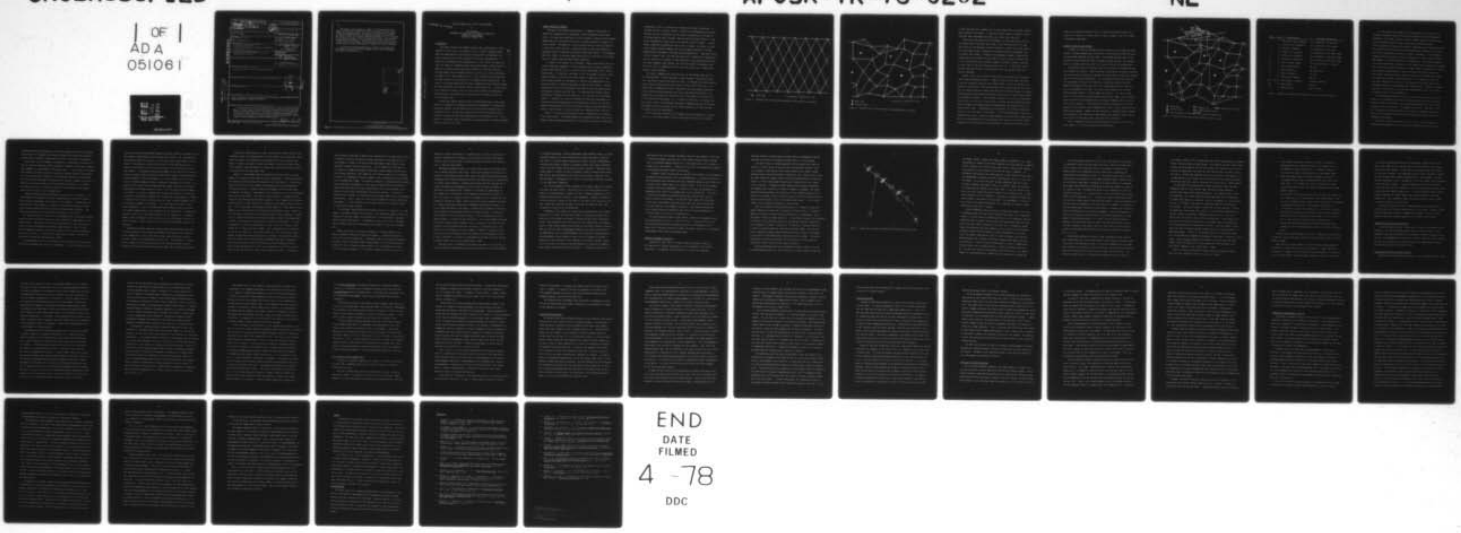
AD-A051 061 OREGON STATE UNIV CORVALLIS DEPT OF ELECTRICAL AND C--ETC F/G 17/9
ADVANCED FORWARD AREA TACTICAL RADAR NETWORK.(U)
1977 J D SPRAGINS AFOSR-77-3339

UNCLASSIFIED AFOSR-TR-78-0202 NL

OF
AD A
051061



END
DATE
FILMED
4-78
DDC

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFOSR-TR-78-0202	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): ADVANCED FORWARD AREA TACTICAL RADAR NETWORK	5. TYPE OF REPORT & PERIOD COVERED: Interim rept.
6. PERFORMING ORG. REPORT NUMBER	
7. AUTHOR(s): John D. Spragins	8. CONTRACT OR GRANT NUMBER(s): AFOSR-77-3339 new
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Oregon State University, Department of Electrical & Computer Engineering, Corvallis, OR 97331	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61102F 2304/D9
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Office of Scientific Research/NM, Bolling AFB, DC 20332	12. REPORT DATE: 1977 13. NUMBER OF PAGES: ~~44~~ 46 pp.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

DDC RECEIVED MAR 9 1978 F

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Radar, Network, Tactical Air Control, Data Processing, Data Communications, Computer Networks, Command and Control

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An advanced tactical radar network which is under conceptual development by the Air Force, with an anticipated installation date in the late-1980s, is described and some of the primary design choices are evaluated. The proposed radar network would provide revolutionary improvements in the quality of coverage by tactical radars by taking advantage of current trends in data processing and data communications.

NEW 410 595

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD A051061

DDC FILE COPY

20.

The heart of the network is a number of closely-spaced short-range 3-D radars which communicate with one another to share information about target tracks. All radars in the network are assumed to employ sophisticated data processing techniques to process target reports, with the resulting information communicated to all radars in the network through a grid of communication links. Each radar in the network will maintain a complete file of track reports for all targets seen by the network. The communication links used will also be available for other communications, command and control purposes.

A variety of algorithms which may be used in the network are discussed, and some of the main problems to be faced in developing the network are pointed out.

ADVANCED FORWARD AREA TACTICAL RADAR NETWORK

by

John Spragins
Department of Electrical and Computer Engineering
Oregon State University
Corvallis, Oregon 97331

INTRODUCTION

This report describes an advanced tactical radar network which is being studied by the Air Force. The network is still in initial planning stages, under the responsibility of the Advanced Planning Office, Headquarters Electronic Systems Division (AFSC), Hanscom Air Force Base, Massachusetts. In addition to conversations with representatives of this office, the author has based this report on information about the proposed network learned during the summer of 1976 while he was stationed at Hanscom Air Force Base under the USAF/ASEE Summer Faculty Research Program [1] as well as on work he did during the summer of 1977 under AFOSR Grant No. 77-3339, Architectural Considerations for Radar Networks as a continuation of the Summer Faculty Research work. The report is an interim report under the grant, summarizing results obtained during full-time employment on the project during July and August 1977. A final report, to be completed by June 30, 1978, will also contain results obtained during part-time work, by the author and at least one of his students, during the 1977-78 academic year.

Although this report contains a fairly detailed description of the overall network, its emphasis is on problems involved in development of a suitable communications protocol for the network since this has been the primary focus of the author's work. It should be recognized that conclusions reached here are tentative in nature and that other approaches may be considered as well. Nevertheless, a reasonably detailed description of some algorithms which appear feasible should be useful at this time.

GENERAL OVERVIEW OF NETWORK

The heart of the proposed radar network is a number of short-range 3-D radars which communicate with one another to share information about target tracks. The radars are spaced fairly closely together (a nominal distance between radars might be on the order of 30 miles) to assure continuous coverage of low flying aircraft. Although most of the discussion here will treat the radars as if they are at fixed locations, a large percentage of them should probably be reasonably mobile (to maximize survivability under enemy attack). The assumption made here, though, is that most radars would move only occasionally, with essentially all radars stationary during normal operations.

All radars in the network will be assumed to employ sophisticated data processing techniques (comparable to those techniques used by Lincoln Laboratory's Moving Target Detector, MTD [2], or RADC's Digital Coded Radar, DCR) to process target reports. In addition, each radar site will be assumed to contain a local processor capable of utilizing the target reports it generates from its own radar returns together with information it receives over communication links from other radars, to generate and update system wide track reports. General descriptions of the type of information that might be used for generating and updating track reports are given later. New algorithms for tracking targets, significantly different from any currently in use, will need to be developed, however. A contract for a study to determine and define the requirements for track-associated data processing was being negotiated by ESD at the time this report was being written [3], with the contract awarded to General Research Corporation, Santa Barbara, CA, while the report was undergoing final revisions.

Figure 1 is a sketch of one possible "ideal" configuration for a portion of the radar network. The diamond-shaped grid configuration shown could be the nominal network configuration aimed for in a particular situation, or some other

configuration, such as a triangular or hexagonal grid might be preferred. In an actual tactical situation, however, a regular grid configuration of the type sketched will not be achievable (due to terrain irregularities, accessability of various possible radar locations, enemy actions and similar factors), so the actual configuration is more likely to be similar to that in Figure 2. (Nevertheless, idealized configurations such as that in Figure 1 are useful, at least at this stage of network planning, for initial rough calculations of network communications requirements and target tracking capabilities.) Important factors to note in Figures 1 and 2 are that each radar site is connected by communication links to several of its nearest (or more accessible via the communication facilities used) neighbors. Both Figures 1 and 2 are drawn under the assumption that each site is able to communicate with at least four other sites, a number which currently seems reasonable.

The type of communication links which will be used between nodes have not yet been specified. Current candidates include microwave, troposcatter, satellite, and packet or spread spectrum radio links. In addition, other possibilities such as optical fibers could be utilized for at least a few of the links. The primary type of communication links visualized by the author during preparation of this report has been microwave radio links using narrow-beam antennas (one antenna per link at each site) since this type of link seems to be feasible and to have definite ECM advantages (i.e., it should be difficult to jam effectively). Very little, if any, of the material following is dependent upon use of this type of communications links, though, and all of the possibilities listed, and some not listed, should be investigated.

One of the primary functions of the communication links indicated in Figures 1 and 2 is to send track reports back and forth among the radars in the network

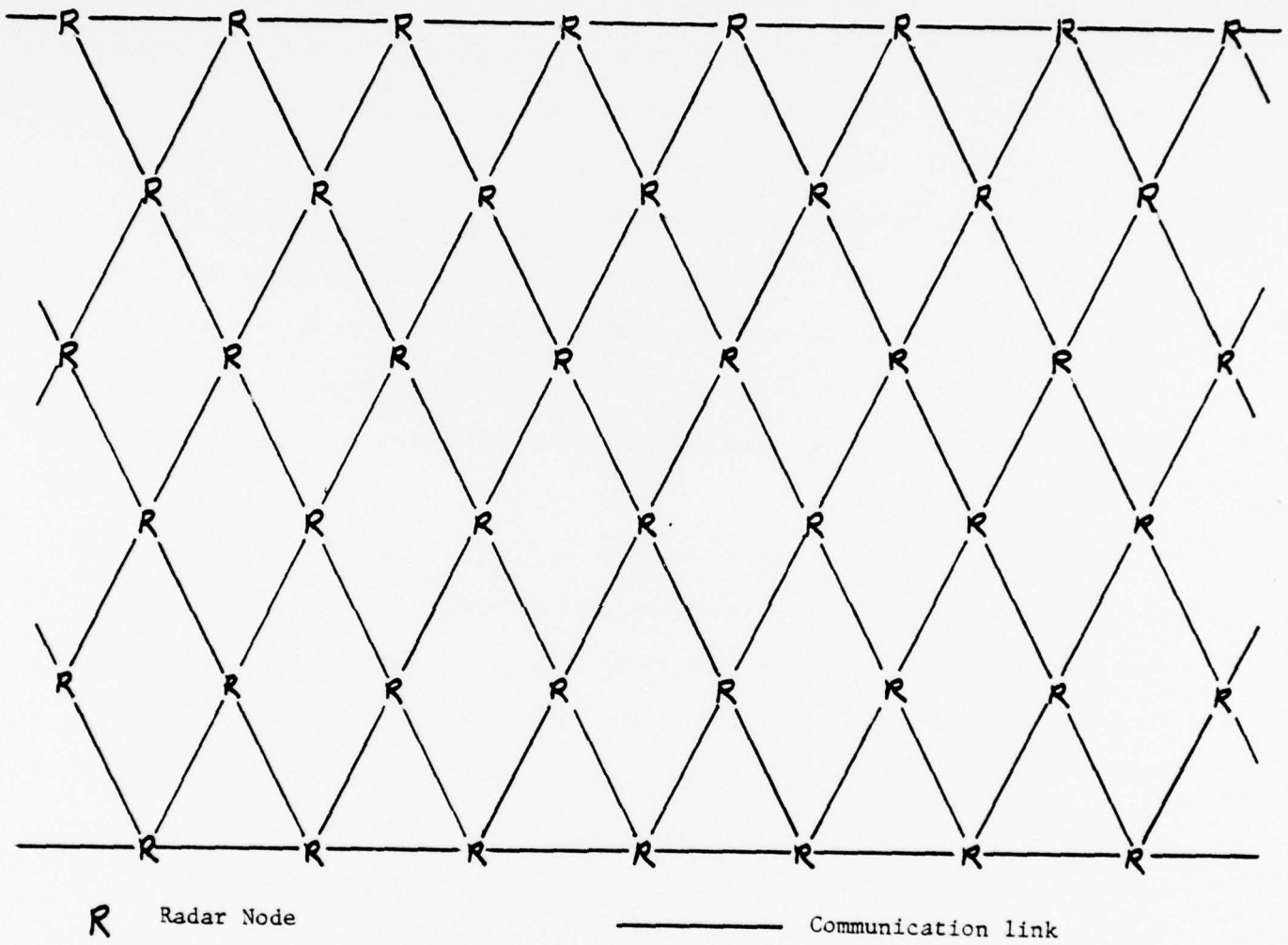


Figure 1. Idealized form of six-level radar network using diamond grid pattern.

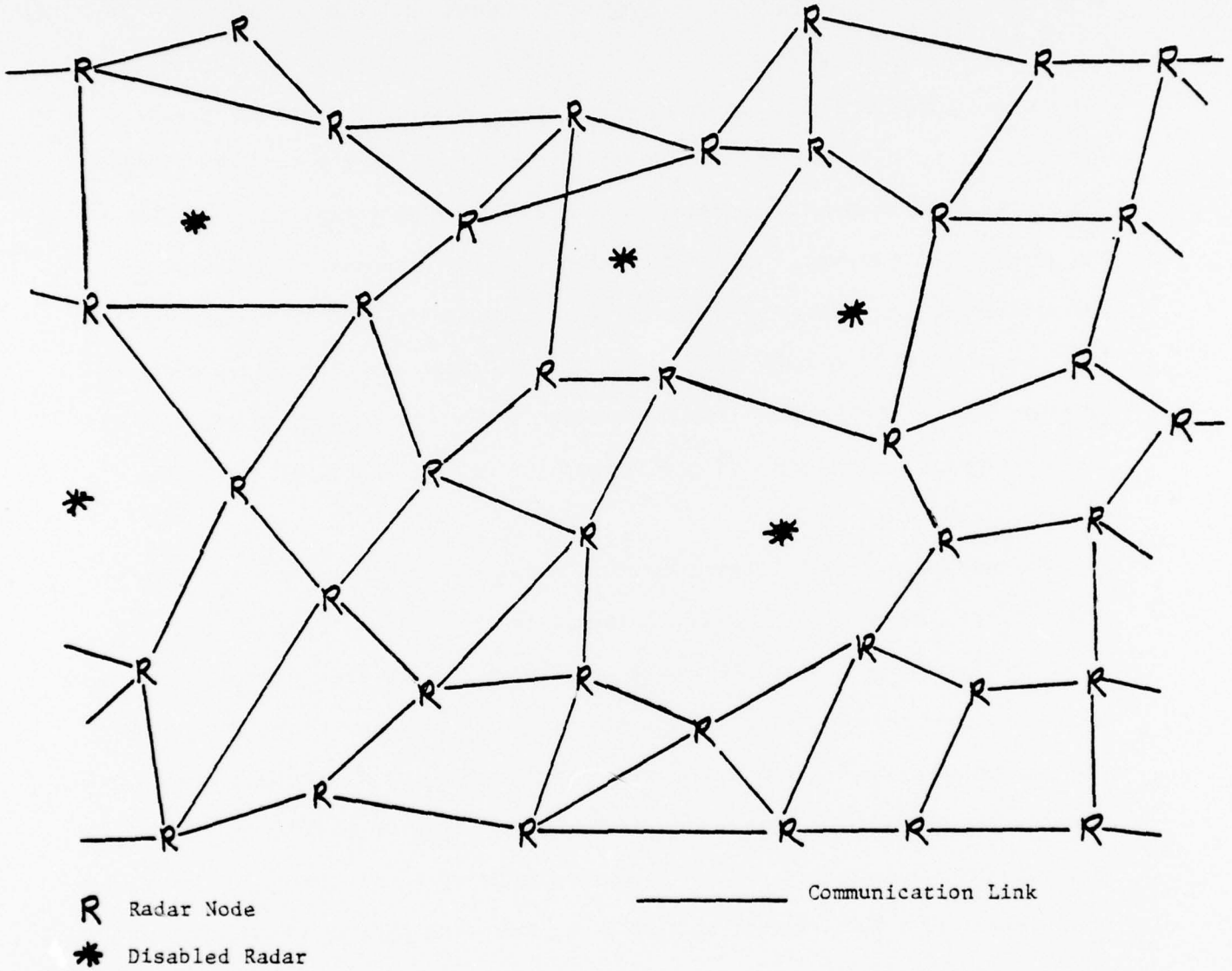


Figure 2. Possible form of radar network with randomized radar locations.

so that each radar has a complete file of track reports for all targets currently being tracked by the network. A large part of this report is devoted to discussing some simple algorithms for passing such reports back and forth.

Since the netted radars are to be primarily in the forward area (close to the Forward Edge of the Battle Area, FEBA), their functions would be in some ways similar to the functions of current Forward Air Control Post (FACP) radars. The netted radar system is expected to be far more flexible and powerful than the current FACP, though. In addition to performing a number of functions not accomplished by any elements of the current Tactical Air Control System (TACS) [4], the netted system would perform a number of functions currently performed by other levels of the TACS hierarchy besides the FACP. In particular, most of the more important functions of the current Control and Reporting Post (CRP) would be performed.

Figures 1 and 2 only illustrate radar nodes in the network and the communication links interconnecting them. A number of other types of nodes, which may or may not be directly associated with radar systems, can also be expected to be attached to the network. These include nodes which exercise some type of control over the network (especially during time periods when nodes are added to or removed from the network--during other periods the current goal is to make different nodes as autonomous as possible). Other elements connected to the net may correspond to other levels of a TACS type of architecture, including various command and control types of elements, up through centers corresponding to the current Tactical Air Control Center (TACC). Interconnection of the network with airborne systems such as the E3-A will also need to be provided. If the system developed has sufficient communications capacity, there are strong arguments for using network communications for still other purposes such as relaying data from some type of Aircraft Control elements out to distribution facilities actually in contact with aircraft on the

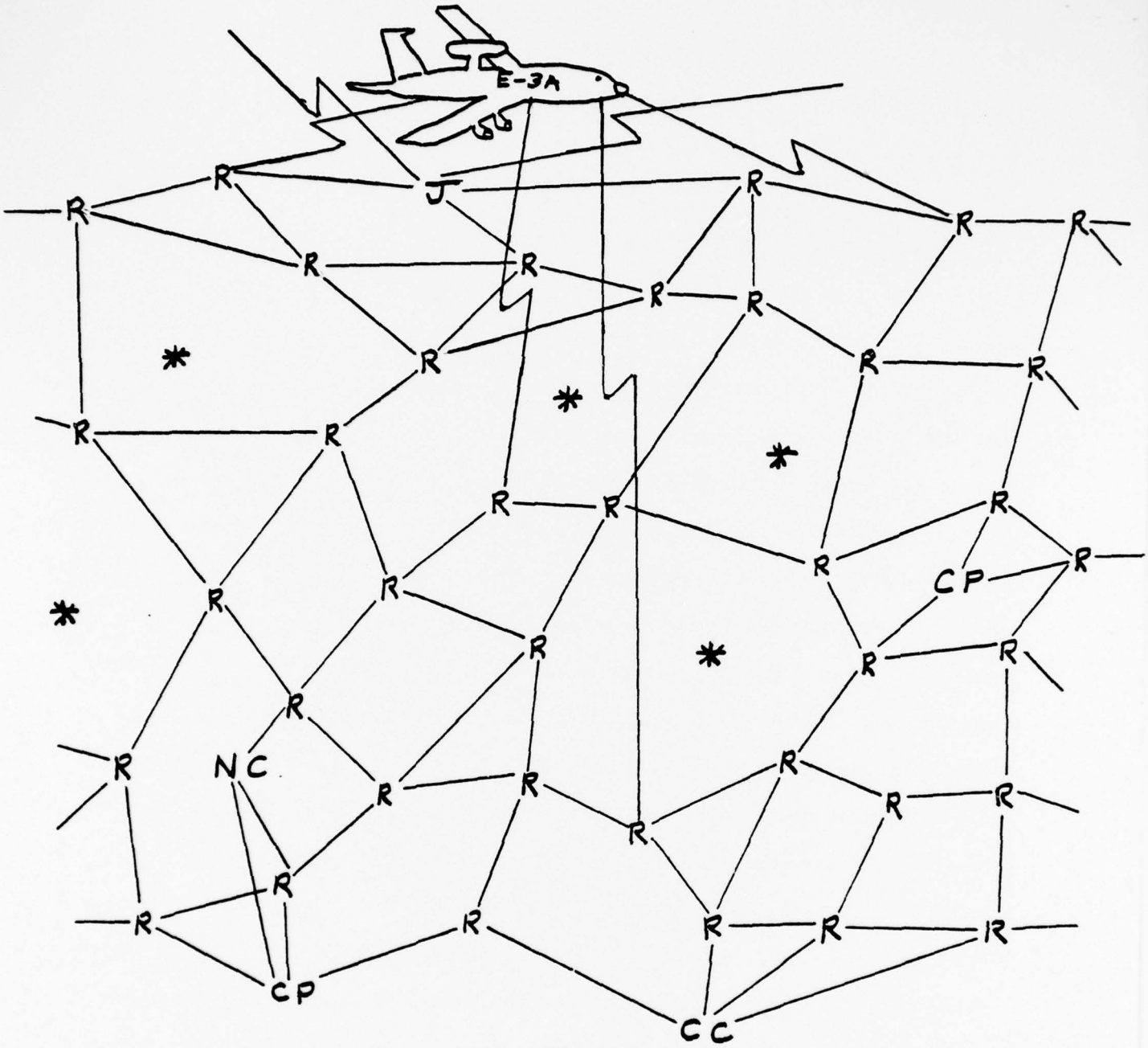
other side of the FEBA (possibly JTIDS, or similar distribution stations [5]). Figure 3 is a sketch of a system similar to that in Figure 2, but with a number of additional nodes added.

PROJECTED FORM FOR TRACK REPORTS

A variety of types of information can be expected to flow through the radar network, including messages sent from certain of the specialized nodes discussed in the previous paragraph to other nodes. The different message types can be expected to require different types of message handling routines, with information in a message header utilized by the system to identify the message types.

The primary focus of the network study so far has been on dissemination and use of radar track reports. Hence, more work on identifying the types of information which should be contained in messages has been devoted to radar track messages than to any other types of messages. Also, the number of radar track messages sent throughout the network will probably be considerably greater than the numbers of messages of any other type. The author suspects that it will turn out to be reasonable to fit most, if not all, other messages into a message format designed primarily to handle radar track reports. (At least he hopes that it will be possible to keep important parts of the format, such as message length, locations of corresponding parts of header information and of error protection data, constant.) There appear to be definite advantages, in a system designed to provide highly secure communications links, to data formats with constant message lengths. Padding out short messages, or grouping several together for transmission, and breaking up longer messages into shorter "packets" can be used to force all messages to obey a constant length constraint.

Table 1 summarizes some preliminary estimates of required field sizes in track reports. All fields indicated are discussed below.



R Radar Node
* Disabled Radar
J JTIDS Distribution Station
CP Command Post
CC Command/Control Center
NC Network Control Center
E-3A Airborne Command Center

Figure 3. Possible form of network with additional types of nodes.

Field	Bits	Identification	Descriptive Parameters
x	16	"Horizontal" grid coordinate	50 m resolution, 1600 km range
y	16	"Vertical" grid coordinate	50 m resolution, 1600 km range
h	12	Elevation coordinate	15 m resolution, 60 km range
$\dot{x}$	10	Velocity in x direction	5m/s resolution, $\pm$ 2500 knots range
$\dot{y}$	10	Velocity in y direction	5m/s resolution, $\pm$ 2500 knots range
$\dot{h}$	10	Velocity in h direction	5m/s resolution, $\pm$ 2500 knots range
t_r	18	Time of current report	1/64 sec resolution, 1 hr range
i_t	24	Track identification	See text
i_r	8	Radar identifier	256 possibilities
b	32	Beacon information	See text
q	32	Track quality measure	See text
misc	50-100	Miscellaneous information	Currently undefined
hdr	32	Report header	See text
tlr	16	Report trailer	Error checking

Table 1. Suggested information and overhead fields for radar track reports.

All track reports are assumed in this discussion to be in terms of "network-wide" coordinates such as latitude and longitude (or other rectangular) coordinates, plus height, and their rates of change, instead of being in terms of local coordinates such as range, azimuth and elevation (and their rates of change) which would be more directly measurable by the individual radars. This implies that one of the data processing functions performed at each radar site is this translation from local to "global" coordinates. (The coordinates assumed are not truly global, since only enough bits to locate targets within the area spanned by the radar network will be assumed here. Additional transformations, with corresponding increases in field lengths, may be necessary for information relayed back to areas interested in larger theaters of operations.) Performing coordinate transformations at individual radar sites before sending out track reports should considerably simplify correlation of network track reports with detections obtained at the other radars, as it eliminates any need for each radar keeping precise records of the locations of all other radars then performing appropriate transformations on incoming reports. Further, it reduces a possible N duplications of the same data transformation (with N radar sites in the network) down to one transformation.

In estimating the number of bits to be used for each data field in a track report, it is reasonable to allow for somewhat greater resolutions and ranges than are currently achievable, since the anticipated installation date of such a network is in the late-1980's. The suggestions here are designed to minimize the probability that the network will, during its lifetime, be unable to fully utilize the capabilities of new technology because of being locked in by inadequate data formats.

The spatial coordinates used here will be x and y (for two rectangular coordinates cooresponding either to latitude and longitude or to some similar

coordinate system better adapted to the network coverage area) and h (for height). The possibility of compatability between the network grid system and other grid systems such as the WGS-72 standard used by JTIDS [5] also merits investigation.

Both x and y will be assumed to be defined to a resolution of 50 meters (note that this does not necessarily imply accuracy of 50 meters), and to have a range of 1600 kilometers (or approximately 1000 statute miles). If the origin of the coordinate system is carefully located with respect to the area covered by the network (say, at the "lower left" corner with all coordinates positive or at the center with both positive and negative coordinates and a sign bit included in the data), a total of 15 bits would suffice to give this range and resolution. In order to give more flexibility in locating the origin of the coordinate system, though, 16 bits each for x and y will be assumed here.

Somewhat different range and resolution values hold for the height, h. For this parameter a resolution of 15 meters (approximately 50 feet) and a range of 60,000 meters (approximately 200,000 feet) will be assumed. (In some cases, the height field in track reports will probably be obtained from beacon data which should be more accurate than can be achieved with pure radar techniques). The range and resolution values listed can be obtained with 12 bits for h.

All velocity values, $\dot{x}$, $\dot{y}$ and $\dot{h}$, will be assumed to have resolutions of approximately 5 meters/second and ranges of at least 0-2500 knots (approximately 0-1290 meters/second). Each velocity can then be represented by 10 bits, including one sign bit, since the target could be moving in any direction. (Use of 9 bits would be marginally possible, but 10 bits is more likely to allow for advances in the state of the art, or it could allow a range of up to almost 5,000 knots if there were any reason for requiring such a range.)

Time measurements must also be included in the data reported if the location and velocity parameters are to be really meaningful. At least one time field, giv-

ing the time at which the reported observation was made, should be included in each track report, and additional time fields may well be useful. For the purposes of the discussion here, one time field, t_r (the time of the current report) will be assumed. (One additional time field, the time of initial detection will be assumed to be maintained in the file for each track, but it need not be sent with each message.) A reasonable resolution requirement for the time field is that the accuracy of target location projections should not be limited by the resolution of the time parameter in the track reports, i.e., other factors, such as the resolution of location parameters should be dominant in determining the accuracy. In order for this to hold true at maximum projected target velocities of 2500 knots (approximately 1290 meters/second) at least 2^{-6} or 1/64 second resolution appears to be needed. (This gives a maximum time resolution contribution to target location error of approximately 20 meters, with a mean error of approximately 10 meters, in comparison with the basic resolution for the x and y location parameters which was previously assumed to be 50 meters. It is conceivable that errors in the height parameters, h, due to time resolution might exceed the basic h resolution of 15 meters assumed earlier, but increasing the time resolution to eliminate this possibility does not, to the author, appear to be very important.) A reasonable range in time to assume would be at least one hour, or 3600 seconds (slightly less than 2^{12} seconds). Hence, a total of 18 bits for the time parameter seems to be reasonable.

Although the one-hour range for time postulated should be adequate for most tactical situations, certain locations connected to the network (which might be called command posts, CPs) may well want to keep longer term records, possibly on disks or tapes, and reference these records later. This possibility is easily handled, though, by additional time labels on the long-term records, which can locate the time period within any desired time range, even centuries-long periods.

Some type of identification, or label, for each track report would help considerably in many of the applications of the data which can be visualized. Hence each track report in the network will be assumed to have its own unique track number. Actually, assigning a unique track number is one of the trickier steps in the protocols being considered, for reasons that should become obvious during the discussion of the track number field.

Normally, a track number should be assigned by the first radar to see the target, or at least the first radar to issue a track report. (Possible techniques for reconciling conflicts here are discussed in the next section.) The most economical use of bits for track numbers would require that each track number be selected from a pool of currently unused track numbers, with each radar using the same pool of numbers. There appears to be no good way for all of the autonomous radar sites in the network (which observe targets coming in at random times and velocities and arriving from arbitrary directions) to use a common pool of numbers without conflict, so each radar will be assumed to select numbers from its own subpool. This is equivalent to prefixing each track number by a radar number, corresponding to the identification of the radar first reporting the track. Eight bits for this prefix (allowing up to 256 different radars) should be generous. Another 8 bits to identify the track itself should be enough to handle the maximum number of tracks in the network at one time with the same initial reporter. (It would be worthwhile to check this assumption by a study of the worst credible circumstances for its validity to hold, however, since circumstances when it would not be true can be visualized.) This gives a total of 16 bits for the basic track number, but additional subfields could still be needed. The additional subfields would help handle some of the more complex phenomena which must be managed by the tracking algorithms used in the network-splitting and merging of tracks. Both splitting and merging will occur, with a possibility that both may have occurred in the past history of what is now a single track report. No attempt to

define suitable algorithms to handle splits and mergers will be given here (in fact, no attempt to precisely define any tracking algorithms is being made as this is more properly a subject of the data systems requirements study currently being conducted by General Research Corporation). For the purposes of this report it will simply be assumed that adequate algorithms to handle mergers and splits and for attaching appropriate track numbers to the resulting tracks will be developed. An extra 8 bits appended to the track number to help in handling splits and mergers will be assumed here, though. This should be adequate to indicate whether the particular track report corresponds to a split and/or a merged track and how many different tracks have been split and/or merged before this one was obtained, but it would not allow for full identification of the earlier tracks. (If such full identification is needed, the length of each track report might have to be increased substantially.) Thus, a total of 24 bits will be assumed for the track identification number, i_t. Track numbering involves enough subtleties to indicate a need for additional research, however.

A considerably simpler identification number, which will be denoted i_r, is the number of the radar issuing the track report. (The radar whose number is implicitly contained in the track number, as discussed above, is not necessarily the radar making the most recent report.) In keeeping with the use of 8 bits to represent possible radars in initially establishing track numbers, 8 bits will be assumed for i_r.

Another field which is difficult to quantify is a field, denoted b, for beacon information (which may or may not be present). An arbitrary length of 32 bits for b will be assumed here. Considerably more bits for beacon information are used in some of the Radar-to-ROCC message formats in the Joint Surveillance System (JSS) [6], but these JSS formats appear to be very inefficiently coded, so 32 bits should be adequate. (An alternative scheme which might be used is to

maintain all beacon information in a separate file, with only an indication of whether a beacon return is present or not contained in the main track report. Then only one bit need be used, to indicate presence or absence of a beacon return, in the main track report.)

Additional information fields in a typical track report are even more difficult to define precisely at this stage. Some type of measure of the quality or accuracy of track reports could be very useful, and sophisticated data processing techniques such as those associated with MTD or DCR should be able to produce some very useful quality measures. Numerous possibilities for such measures can be listed, including the percentage of the last N scans on which the target was detected by the reporting radar, strength of the returns or some measures of the dispersion of these returns (brief examination of some actual reports of MTD by the author [7], during his work in the summer of 1976, convinced him that patterns in the multiple range, azimuth and velocity returns obtained from scans of a single target contain much useful information about detection quality as well as about the nature of the target), or some types of estimates of the sizes and orientations of the ambiguity regions for target location. A fairly arbitrary length of 32 bits for a quality field, q, will be assumed here, but the required number is a subject of some controversy. The 32 bits assumed here would allow quite elaborate quality measures, such as estimates of the sizes of error ellipses in each coordinate direction, but there is a strong possibility far fewer bits will be needed for a quality measure. Hopefully, the quality field will give a realistic measure of the true accuracy of the measurements, and be in such a form that it can readily be combined with new input data from the same or another radar site to give an updated quality index.

Additional types of information that might be included in track reports include more detailed information about detections obtained, such as actual doppler

or strength measurements. (Doppler measurements from different radars, plus some information about the relative orientations of radars and targets, should be very useful in tracking targets taking evasive action, possibly unamibiguously indicating such things as whether two targets actually cross paths or approach closely and then diverge from one another without crossing.) Information about strong sources of clutter or other disturbances (readily extracted from clutter maps such as those maintained by MTD) should also be useful, especially in an ECM environment. Approximately 50 to 100 bits (or even more) might well be used for such additional information.

All of the information fields discussed so far contain some type of information about the target tracks. Additional fields in the track reports will be needed for overhead purposes, primarily to help in handling communication of information among different locations. Such information is normally included in two main locations, a header (at the start of the report) and a trailer (at the end), though other locations for parts of the data are sometimes used. Both a header and a trailer will be assumed here.

Information in the header will include data indicating what type of transmission this is (recall that many other types of transmissions besides track reports will flow through the network), synchronizing and control information, possibly some information about the priority of the transmission, any sequence counts or other message numbers that are used to keep track of messages, possibly some bits that are used to acknowledge information sent from the other end of the communication link (if acknowledgements are required and "piggy-backing" of acknowledgements on data messages is used), plus any information relevant to congestion or other measures of the conditions at individual locations (which can be very useful for routing of directed messages or allocation of communications capacity between flows in two directions). Although the functions to be

performed with the aid of header information should be much simpler in this type of specialized network than they are in more complex and more general networks such as the ARPANET computer network [8], a fair number of bits will probably be needed even in this network. A figure of 32 bits will be assumed for the purposes of developing preliminary report size estimates.

Although a variety of functions for trailers can be defined, only one function, transmission error protection, will be considered here. It should be possible to provide adequate error protection with fairly moderate numbers of error check bits if these are only used to detect (not correct) errors. If one of the standard polynomial codes with approximately 16 check bits is used, the probability of undetected errors should be low enough that it can be ignored [9]. (LSI chips to perform the encoding and decoding functions for some of these codes are readily available from vendors.) On the other hand, actual error correction in the type of environment anticipated here, where error rates could become very high (especially when jamming is going on), would require far more overhead, probably requiring considerably more check bits than information bits. A trailer length of 16 bits will be assumed here.

The total projected length of track reports, based on the assumptions given above and tabulated in Table 1, is 286 to 336 bits. A value of 320 bits will be assumed whenever necessary to make calculations in the remainder of this report, and in additional research currently being done, since this is a reasonably round numbered value within the projected range.

POSSIBLE APPROACHES TO TRACKING

Although this report does not discuss tracking algorithms in depth, a few suggestions on tracking algorithms which have been made will be briefly treated here. The emphasis of the discussion is to point out important

problems related to tracking which the author feels are fundamental problems that must be solved if a satisfactory radar network is to be developed.

Two major classes of tracking algorithms for the network have been discussed: a class in which, at any one time, one designated "reporter" radar is in charge of reporting each track being observed by the network (though different tracks could have different "reporters"), and a second class in which any radar observing a tracked object and having something useful to report may report it, with no single radar having explicit responsibility for a target at any given time. Although the former approach would conserve bandwidth in the communications facilities, and might be simpler to implement, the latter approach has more potential for developing a really powerful approach to networking in the future, since it gives the capacility of using frequent looks by multiple non-responsible sites to track maneuvering targets. It will be the primary type of algorithm discussed in later sections.

Part of the reason for the greater potential of a network with multiple radars reporting the same track is indicated by Figure 4 which is a sketch of ambiguity ellipses for target locations obtained by two asynchronously scanning radars tracking the same target [10]. As the figure indicates, the difference in locations of the two radars results in considerably different orientations for the ambiguity ellipses. Intuitively, it seems that it should be possible to devise a tracking algorithm, using inputs from both radars, such that the overall ambiguity along any spatial coordinate would be close to the minimum of the values in this same direction applying for the two radars. The resulting ambiguity figures would probably not be elliptical, but they would be considerably smaller than any of the curves in Figure 4.

Tracking algorithms which utilize information from all radars in a close to optimum manner probably will not be feasible in a radar network of the type

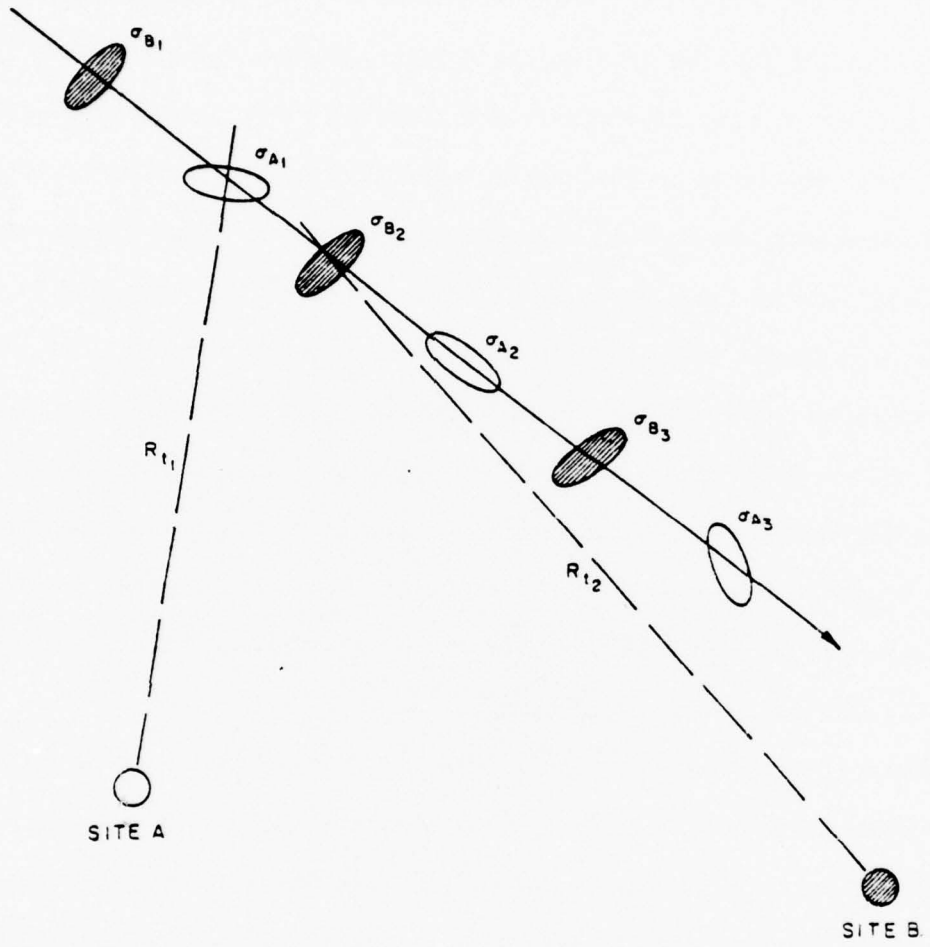


Figure 4. Radar error ellipses obtained with surveillance from two sites.

envisioned, however. Rather than sending complete information on all target detections to a central location which combines the various reports to generate tracks, tracking will be done at each radar location using the best information available at that location at the time the track is computed. The information used would be any tracks reported to that location by other radars, plus measurements obtained by the radar computing the track (if it also sees the target), or it could conceivably include information from two or more distinctly numbered track reports sent in by different radars that have not yet reconciled their reports for the same target, if two such reports arrive within the brief integration period allowable at one location before it must send information on to its neighbors. Algorithms to successfully handle such varied sources of data need to be developed. This development is critical to the success of the radar netting effort if the approach with multiple radars reporting the same track is adopted.

Another fundamental problem related to tracking which needs to be solved is determining when each radar should transmit track reports to its neighbors. Actually, two major classes of reporting functions will be performed, relaying unmodified copies of reports received from other neighboring radars and sending out new track reports generated at the radar site. Although the normal reason for generating a new track report at the site should be to incorporate information about the track obtained by actual measurement at a site which "sees" the target, other reasons (such as a need, according to the tracking algorithm being used, to merge two distinct incoming track reports into one) are conceivable. Simple relaying of track reports is really a communication function rather than a tracking function, so routing algorithms, protocols, etc., for the relay function are discussed in the next section. Only general comments on determining when to transmit new track reports are included here.

If a good tracking algorithm for use in the radar network is developed, it is reasonable to expect that any radar which sees a particular target would be able to improve (on a statistical basis) the accuracy of incoming track reports which do not incorporate its most recent data by generating new reports which do incorporate the data. In the network being studied, though, track reports can be expected to flow among different radars in patterns which are not entirely predictable in advance, so it may be difficult to determine which radars have contributed to generating a given track report at a particular instant of time. (Although each report of new information will have an originator identified, it is unlikely that a list of contributors will be maintained in each track report file.) Limiting each radar to generating no more than one track report update per scan for each target it sees would guarantee that each new piece of information is only used once in this manner. In addition to limiting unpredictable variations in the accuracy of tracking (due to treating identical observations as if they were independent), this would eliminate any possibility of having a single new piece of data cause generation of a large number of updated track reports (i.e., radar A uses its new observation to generate a new track report which it sends to B, then B--which also observes the target--generates a new report sent to A, then A takes this new report to combine with the same observation used just previously and sends a second report to B, etc.).

The proposed algorithm of allowing a radar to generate no more than one track report update per scan for each target it sees has some limitations since a major goal of the proposed network is to allow new information to be disseminated throughout the network as rapidly as possible, while the algorithm implies considerable delay may occur before some new reports may be sent out. The proposed procedure doesn't limit speed of dissemination as much as might at

first appear, though, since no appreciable delay before retransmitting (un-updated) reports from other nodes has been assumed. (Such reports from other nodes should already include information from the local node's last observation, if they are received any appreciable time after the local node has sent its update to adjacent nodes.) Without some algorithm such as that indicated, there appears to be a strong possibility the network will become saturated with superfluous reports. Considerable additional study on techniques for avoiding this is needed, however.

An algorithm for determining when to make track reports which has been suggested by ESD XRT [11] might well make the concerns about generating multiple track reports above become meaningless. The suggestion uses a "dual box" algorithm. Two "boxes" about projected target positions (as determined by current track reports) are defined. Each box corresponds, at a given instant of time, to a volume in space about the predicted location of the target at that instant. (The precise shape of the box has not been specified. Further, the appropriate "state space" may include ID, split and merge characteristics, velocity, doppler discriminants, maneuvering condition, etc.) One box is smaller than, and entirely inside, the other larger box. If the observed target position is inside the smaller box, no new target report is transmitted. If the observed position is between the two boxes, a new track report is made but no new track is defined; the most common cause of this would probably be a maneuvering target. Finally, if the observed position is outside the larger box, the observation is assumed to correspond to a new target (or possibly a false alarm). When an adequate number of such reports to define a new track (and largely exclude the possibility of false alarms) have been obtained, a new track report (numbered by the issuing radar) is made.

Some more detailed guidelines for report generation/redundancy removal, which have been suggested by ESD XRT [11], are as follows:

1. Each incoming track report received by a node is compared for message identity with messages already in the message file. If the message is already in the file, it is dumped; otherwise it is sent to the system track processor as a track-update report.
2. Each track-update report (internally generated or received via the communications system) is examined against the stored system track files. If the report falls within the smaller box about the projected stored track, the track is continued and the message dumped, otherwise
3. If the report is outside the smaller box, but within the larger box, the system track file is updated and a track update report is sent out to all neighboring nodes (except the one from which the corresponding message was received if it was externally generated), or
4. If the report is outside the larger box, it is filed as a tentative new track if the report was received as an update or was generated at that node, or as a definite new track report. The new track report is then sent out on lines not having reported it in.
5. Provide for complete system track file dump upon request from a neighbor (for auto-registration and sign-on) or for rare system-wide checks.

Although these algorithms appear to be useful, numerous questions need to be answered. Hence, considerable additional work on further study of the algorithms is needed.

Some conflicts in track numbers utilized by different radars appear to be inevitable, though they may not be very frequent. At times, more than one radar will transmit a new track report on the same target to its neighbors. This will occur whenever the second radar's report is ready to go out before

it has fully received and correlated the other radar's report with its own data. Also, there will be occasions when two or more tracks will be merged into one for other reasons. An algorithm for selecting unique track numbers in such cases is needed. Such algorithms should be easy to develop if information in the track report formats suggested above is used, however. The most relevant information to use is the quality of the track report, giving precedence to the best quality report. A second information field that could be used if this does not resolve the conflict is one of the time fields, possibly favoring the radar that saw the target first. A final tie-breaking decision could always be based on the radar's number, say giving precedence to the lower numbered one. Thus, reasonably simple algorithms which would always succeed in resolving conflicts can be defined.

The material given is far from a complete analysis of tracking, but no further discussion of tracking algorithms will be given here. The next section discusses some ideas relevant to communications protocols, which have been emphasized more than tracking algorithms in the author's work to date.

COMMUNICATIONS PROTOCOL CONSIDERATIONS

As has been stated earlier, a wide variety of types of information can be expected to flow over the communications links in the radar network. Some reasonable techniques for handling such information flows are discussed in this section. The recommendations here are still tentative, but they tend to be more precise than those made in most other sections since communications protocols have been a major focus of the author's work.

Synchronous Transmission Schemes Suggested

Communication between neighboring radars, or other neighboring nodes in the

network, will be assumed to occur in a synchronous manner, with all information sent in fixed size data blocks (which will probably approximate 320 bits in length, according to the discussion of track report lengths given earlier). (Synchronization need only be for each individual link, not system wide, as all links operate independently of one another.) The block length used will be assumed to be selected primarily to fit the requirements of radar track reports since the number of such reports transmitted should greatly exceed the numbers of any other types of blocks sent through the network and the projected lengths for radar track reports appear to be reasonable for most other types of transmissions. If necessary, long transmissions can be broken up into several successive blocks or packets, and short transmissions can be padded out to the appropriate length or (preferably) several short transmissions can be chained together to form one packet.

The preferred type of synchronous transmission in a tactical environment is probably completely synchronous transmission, i.e., successive packets follow one another over the communications link without intervening pauses, even if some of the packets transmitted are empty (which would be indicated by a header). (Encryption might or might not be used, but this can largely be ignored in the current discussion since it would have little impact on data rates and communication protocols. The main impact would be on the data processing requirements for encryption and decryption at each end of the link.) The completely synchronous mode of operation suggested simplifies clocking requirements at each node, which can lock onto the pattern expected and hold onto it reasonably well even in a very hostile environment with heavy jamming. Some penalty is paid in the sense of having unused packets transmitted (though nothing is really lost when the extra capacity would otherwise be unused) and the possibility of slight extra delays for data that comes in for trans-

mission, but the extra delays can be expected to be minimal and no significant loss of capacity due to unused packets will be observed under heavy loading conditions (when virtually all packets will be used productively). Use of a completely synchronous mode of operation also eliminates the need for using extra bits for "framing" (establishing message synchronization) at the beginning (and possibly the end) of each transmission. Hence, no bits for framing were included in the track report formats discussed earlier. Further, a major advantage of completely synchronous operation in a tactical environment is that it makes intelligence work by an enemy more difficult; in particular, no useful information could be obtained by monitoring the amount of "data" transmitted throughout the network wince the volume would remain constant regardless of whether useful information was contained in the packets or not.

The completely synchronous mode of operation suggested has some slight similarity to synchronous time division multiplexing (STDM) [12], but the comparison is not too close since a particular slot is not dedicated to a particular conversation. In the radar network the majority of the messages sent are expected to be single packet messages, so there is no need for a long term dedication of a portion of a link's capacity to handling any particular message. Except for the fact that the starting and ending times of packets (including some packets that may be empty) are precisely determined, the completely synchronous mode is even more similar to asynchronous time division multiplexing (ATDM) [13] since an incoming message always utilizes the first packet time slot occurring after it arrives (if there are no messages queued up) or after it makes its way to the head of the waiting line. STDM, ATDM and the completely synchronous mode of operation suggested are all different forms of time-division multiple access (TDMA) [14].

Some modifications to the completely synchronous mode of operation will be necessary if the communications paths in the network are not full duplex (i.e., capable of transmitting information in both directions simultaneously). A typical case where full duplex operation would not be possible is when one antenna is used for both transmitting and receiving, with all transmissions falling within the same frequency band. Use of adequate time and/or frequency diversity to provide full duplex channels may be feasible, but satisfactorily allocating time and frequency slots among all nodes in a network of the type anticipated may pose formidable problems. Hence, full duplex channels may not be feasible. No adequate study of the problems to be expected in satisfactorily allocating channels for full duplex operation has been made.

If only half duplex (one direction at a time) communication channels should have to be used, an "almost completely synchronous" mode of operation is indicated, with reversals of the direction of information transfer occurring at instants of time known to both ends of the transmission links. (This mode is less desirable than a completely synchronous mode, with one of its major limitations being inferior traffic security.) Since the relative volumes of information transfer in the two directions will often differ, and will change with time, techniques for allocating communications capacity between the two directions are needed. If information about current congestion conditions (e.g., the amount of buffer in use or similar measures of congestion) is included in the packets sent, appropriate information for allocating capacity between the two directions should be available at both ends of the link. Algorithms for determining this capacity allocation should be easy to define. One caution that should be observed, however, is that a node should not unilaterally change its mode of transmitting and receiving without verification that the other end is going to make corresponding changes in its behavior. Either a message signalling the change must

be sent and acknowledged or information automatically causing the change to occur (e.g., a particular type of congestion situation) must have been sent and acknowledged before the change occurs. (Even under these conditions, loss of the required acknowledgments could initiate interesting error recovery situations.)

With either full duplex or half duplex channels normally used, there are likely to be occasions where a receiving node needs to turn off its transmitters for a period of time (using blinking or decoys or because of a move), but still needs to receive update information, to keep its system file current. In such cases, simplex (one-way) operation may be routine for a period of time.

One other important problem that must be solved in selecting the appropriate communications mode is interference from communications links other than the ones connecting a particular node to its neighbors. Conflicting predictions on how much interference is likely are currently being made, but the author suspects it will be a problem for a few of the links at least during some periods of time. Possible methods for handling it include using different frequency bands or time intervals for transmission on different links, using highly directional transmitting and receiving antennas, and using different encoding and decoding sequences (if spread spectrum techniques are used) [15].

Major Classes of Data Transmissions

Although it is currently impossible to itemize all classes of data transmissions (due to inadequate definition of the total system), a few major classes can be listed.

One type of transmission, which has already been discussed, is that of radar track reports. Such reports will doubtless be by far the most common members of a class which might be called non-directed transmissions, since they

are not really directed toward any particular nodes. Non-directed transmissions will normally be expected to disseminate essentially the same information to all (or virtually all) nodes. In addition to track reports, a number of other types of messages are expected to use this mode, but no list of such message types is available yet.

The majority of the other messages besides track reports handled by the network will probably be directed messages sent from one particular node to another node specified in the header for such messages. (Routing mechanisms to insure that such messages are directed correctly are covered in a later subsection.) Typical uses of directed messages will be for network setup or reconfiguration (which is expected to be handled via directed messages sent between a network control node and specified additional nodes in the network) or dissemination of command and control information to aircraft or weapons systems and reception of information from such systems (which are expected to be handled via directed messages between some types of command and control posts or centers and appropriate information dissemination nodes, possibly nodes using JTIDS links for communication across the FEBA). Numerous other messages to perform various functions of the current TACS [4] could be listed.

Most, if not all, of the directed transmissions are expected to require definite confirmation they have been correctly received at their destination. Error detection coding, with either positive acknowledgements or both positive acknowledgements and negative acknowledgements plus retransmissions (after no acknowledgement is received within a specified time-out period in the former case or a negative acknowledgement is received in the latter case) should adequately handle the required confirmations.

Since some types of messages in the network are certain to have more critical time constraints than others, at least a limited degree of priority structure

should be incorporated. At minimum, this should allow for two types of data flows, a "regular" or "routine" flow and an "expedited" or "priority" flow, with expedited messages waiting at a given node transmitted before regular messages destined to go out over the same link.

The different types of messages, their priorities, the addresses to which they may be directed and other needed information will be contained in a header at the beginning of the packet.

Error Handling Mechanisms

The primary error control mechanism planned is error detection, with retransmission of (at least the more critical) erroneous messages. This approach is almost universally accepted for data transmissions over burst noise channels such as most radio channels or even telephone channels [16]. Effective error control for such channels is also possible with fairly powerful error correcting codes, such as rate 1/2 or 1/3 convolutional codes [17], but such codes typically require more than 100% overhead (100% overhead and 200% overhead for the rate 1/2 and 1/3 codes, respectively) in order to be effective for channels of the general type envisioned here. Conversely, very effective error detection is possible with far less overhead for check bits [9] (the 16 check bits out of 320 bits per packet assumed earlier represent 5% overhead). So long as errors and consequent retransmissions are not too frequent (more than 1% of the messages retransmitted should be unusual), the total overhead for error detection and retransmission should be far less than that for error correction. In addition, the decoding equipment for error detection is far simpler than that for error correction which can require the equivalent of fairly sophisticated computers at each site for performing this function alone.

Error detection and retransmission imply the use of either positive acknowledgements alone, with retransmission occurring if no acknowledgement is received within a specified timeout period, or both positive and negative acknowledgements, with retransmission if a negative acknowledgement is received [18]. Both types of acknowledgement schemes are in common use in currently existing systems.

One subject of debate in the current literature is whether acknowledgements should be handled on a link-by-link basis or on an end-to-end basis, [19] i.e., whether verification of successful transmission across each individual link in a path traversing several nodes or simply verification that the message has reached its final destination should be used. A hop-by-hop acknowledgement scheme is currently favored (by the author) for the radar network, with occasional end-to-end acknowledgements also used for those cases where verification of reception of messages by the intended receiver is most critical. As the discussion of possible acknowledgement schemes given here indicates, it is possible to provide hop-by-hop acknowledgements with very low overhead requirements for acknowledgements if acknowledgements are "piggy-backed" on data transmissions. Such acknowledgements become almost free and should be adequate for most purposes. Also, hop-by-hop acknowledgements imply that an intermediate node along the path has a more clearly defined period of responsibility for a message, beginning when the message is first received and ending when acknowledgement of successful receipt by the next node comes in. This implies that the nodes can operate more nearly independently of one another. The choice between hop-by-hop and end-to-end acknowledgments has not been firmly made, however.

Although the error detection and retransmission schemes listed should take care of the vast majority of the errors that occur, there will be occasions when more complex error handling routines become necessary. Typical cases when such routines are needed are cases when multiple attempts at retransmission are un-

successful or when important reply messages are missed (the acknowledgement scheme outlined below will require special error recovery routines in rare cases, for example). Development of adequate error recovery algorithms is normally the most complex part of developing a communications protocol, and no attempt to fully develop such algorithms will be made here. A few useful guidelines to such a development will be given, though [20-22].

One of the most important characteristics of successful error handling algorithms is that they must be complete; i.e., they must handle all possible circumstances. The only real way to insure that the algorithms are complete is via an exhaustive listing and evaluation of all possible situations. Both graphical and tabular forms of such listings are possible, although the graphical forms (which are similar to detailed state diagrams) may be easier to interpret. For each possible state of the system (determined by its past history), and each possible new input (i.e., message received correctly or incorrectly, or even missed), the next state and any output (i.e., messages sent) must be listed. The behavior of the system can then be traced through to verify that it will successfully handle all foreseeable conditions. Although the procedure is tedious, it is the best procedure known for verifying that a protocol is complete.

One additional point that should be mentioned is that the location in the network which is responsible for error recovery in a particular situation should always be predetermined, with essentially no conditions under which transmission of a particular message from one location to another causes a shift in responsibility. The motivation for this suggestion is some problems which are encountered with IBM's Binary Synchronous Communications (BSC) [23] in circumstances where transmission of an EOT character from one location to another transfers responsibility for error control. (In BSC's terminology, this transmission causes an interchange of MASTER and SLAVE status between the two ends of the link.) A lost

EOT occurring under these circumstances is probably the most serious error that can occur in a system using BSC.

Routing Algorithms

Routing of nondirected transmissions, such as track reports, should be relatively simple. An algorithm suggested by ESD [11] is so simple and powerful that the author has not been able to appreciably improve on it. This is to have each node which receives a nondirected message relay the transmission on to all neighboring nodes save the one (or ones) from which the message came. It is trivial to verify that this algorithm will quickly disseminate such messages to all nodes to which communication paths exist. Some possible slight modifications to this algorithm which the author considers likely are for cases where a track report is being relayed throughout the network and the radar at the relaying node also sees the target. Under such circumstances, the node may need to update the report before relaying it, but resolving whether this should be done or not should be part of developing suitable tracking algorithms for the network.

Routing of directed transmissions should be slightly more complex, but another algorithm (also suggested to the author by ESD [11]) appears to give a simple way of handling this. The algorithm assumes each node maintains records of the links on which incoming nondirected messages originating at other nodes were first received. Such links should be the initial links in the shortest (i.e., quickest) paths for sending directed messages to the nodes which originated the nondirected messages. Hence, directed messages can normally be routed from node to node along the best path by providing tables in memory at each node indicating which outgoing link is best for messages to each possible destination. Any changes in network configuration, capabilities, or persistent congestion along certain paths will automatically be handled by changes in these tables which occur when

nondirected messages arrive via different routings.

The major potential weakness of this routing algorithm for directed messages which has been visualized to date would arise if there are any potential destination nodes which do not originate nondirected messages for long periods of time. Such nodes could include some very important types of nodes, such as the possible Command and Control Post/Center nodes or forward area command and control nodes which communicate with aircraft or other systems via JTIDS or similar means, if essentially all the messages sent out from these nodes are directed. A simple remedy to this problem is to require that any node send out at least one nondirected message of some type within a reasonable time period (ten seconds to a minute or so seems reasonable). Thus, if the node does not have any track reports or other normal types of nondirected messages to send, it would still be required to periodically announce its existence to the rest of the network. The routings by which these messages reach other nodes would then be used to find routings for directed messages.

One other problem which will need to be handled is what happens to directed messages when something happens to disable the paths they are using while they are enroute. Presumably, error handling routines to deal with this problem will be included among the system error routines.

Modifying the Network Structure

One of the more important features of the radar network is expected to be the ease with which it can be modified as the tactical situation changes. Modifications can result from nodes being knocked out by enemy action, from nodes being moved from one location to another (normally close by) location, or from new nodes being brought in (say additional radars being added to fill in network blind spots),

or from other reasons. The communication links used by the network and the routings used for messages will need to be adjusted as these changes occur.

An essential first step in modifying the network structure is to make the operating portions of the network aware of the need for a change. In cases where nodes are removed from the network, this should be fairly simple. Scheduled removals could be announced by appropriate messages sent from the node being deactivated to other nodes with which it is already in communication, while unscheduled removals (where a node suddenly disappears) should soon be discovered when no messages leaving that node are received at its neighboring nodes.

Network reconfigurations resulting from fairly minor movements of nodes should also be reasonably simple to handle, especially if some notice of the planned move has been given to the network. Transmissions from the new location will probably be received reasonably well by at least one of the neighboring nodes which were previously in communication with the relocated node despite the fact that these nodes may be using narrow-beam receiving antennas for communications. (Some planned rotation of the receiving antennas to approximately the correct angles for receiving from the new location can be done in preparation for receiving the first messages if the move is preplanned. This could even handle planned movements over more substantial distances.)

The most difficult changes in network structure to handle should be those where a new node is ready to be added and the operating nodes have no prior notice of this. (In a highly confused tactical situation, this could be the normal case.) An especially difficult special case can occur when all of the logical nearest neighbors for the added node in a reconfigured network already are in communication with the maximum number of neighboring nodes they can communicate with. (Recall that a normal maximum limit on this number, assumed to be four neighboring nodes in diagrams drawn, was assumed earlier.) Developing

techniques for notifying the rest of the network of the need to reconfigure under such conditions will require considerable effort. It will be assumed here, though, that suitable techniques can be developed. (Possible mechanisms include sending out special short messages at high enough power to be heard even on side lobes of the directional receiving antennas, including omni-directional antennas at nodes specifically to receive such "reconfigure" messages, having the new node send in special signals at radar frequency which can be recognized by the radar receiver when it scans that location, sending information to airborne relay stations which can send them on to the network, etc. ECM implications of these possible mechanisms have not been studied.)

Although individual radars are expected to operate autonomously under normal conditions, some type of centralized control over network reconfigurations is needed in order to insure that reconfigurations are done in a globally optimum manner. Hence, information about requests to be added to or disconnected from the network are assumed to be relayed to one or more network controller nodes (which also are responsible for deciding when unscheduled removals occur). In order to insure survivability of the network, a number of nodes should be capable of handling the network controller function, and each of these nodes should be connected to the rest of the network by several communications links. All requests for changes in the network configuration will be made via directed messages sent to the network controller nodes, and the reconfiguration will then be accomplished via directed messages from these controller nodes to other nodes instructing them to perform such operations as disconnecting certain communication links and establishing others.

Presumably the network controller nodes might also have responsibility for coordinating other possible network functions such as "blinking" of radars to confuse anti-radiation missiles (ARMs). Innumerable questions about how blinking

would be managed can be enumerated, such as whether a blinking radar turns off both its radar and its communication links when it blinks (or turns them on and off during different time periods), how rerouting of diverted messages is achieved if and when communication links are disabled in this manner, etc. Such questions are beyond the scope of this preliminary study, however.

A Suggested Acknowledgement Algorithm

Some debate is currently going on over the extent to which message acknowledgements (either positive or negative) are necessary. Since radar track reports can be expected to be updated fairly frequently, little degradation in performance is expected if reports containing errors are simply ignored. (This assumes a good error detecting code is used, but this poses no difficulties.) Neither positive nor negative acknowledgements appear to be very useful for most track reports. On the other hand, positive verification of receipt of some other types of messages is expected to be very definitely needed, so some types of messages will require acknowledgements.

Although both positive and negative acknowledgements (signifying, respectively, that messages have been received correctly or with detected errors) are used in many commercial transmission control procedures, only positive acknowledgements are really required for a successful communications control procedure. (This is verified by the fact that a number of systems are operating successfully with only positive acknowledgements used.) Both positive and negative acknowledgements will be used in the techniques proposed here, but the tradeoffs between this procedure and an approach using only positive acknowledgements merit further study.

Despite the fact that a large percentage of the messages that flow through the network do not appear to require acknowledgements, the author currently

favors an approach which uses positive or negative acknowledgement for each message, but with these acknowledgements "piggy-backed" on other messages flowing in the return direction (in a manner described more precisely below) to minimize overhead. Some of his motivations for using acknowledgements for all messages are as follows: 1) Treating all messages alike in this manner gives a much cleaner and simpler protocol. Although the difference in simplicity of the two approaches is minimal under normal conditions, it will become much more obvious under error recovery conditions, when having to check for several different special cases can become very tedious. 2) When errors do occur, they may occur at unpredictable locations within a message, with the header indicating the message type as likely to be corrupted as any information. Hence, the information indicating whether acknowledgements are desired or not could easily be the information lost when errors occur. (It should be noted, however, that this will only cause problems when both positive and negative acknowledgements are used; with only positive acknowledgements used, an acknowledgement can only be expected to occur when the header is received correctly.) 3) In general, the final decision on whether a particular transmission should be repeated or not (or other error recovery techniques attempted) when an error has been detected should normally be made at the transmitting location rather than the receiving locations, as the transmitting location will have much more complete information about messages. (The information indicating whether the message was critical or not could easily be the information received in error.) Even the types of messages which normally would not require acknowledgements can conceivably become critical under certain conditions (e.g., there may well be a very limited number of track reports which really are critical), and the information determining this is most likely to be available at the transmitting location. 4) If the acknowledgement messages are "piggy-backed" on other transmissions in the manner suggested below, acknowledgement

overhead becomes minimal, so little if any penalty is incurred by sending some acknowledgements which might not appear to be absolutely essential.

The piggybacked acknowledgement scheme suggested here (a tentative suggestion only at this point) is motivated by the piggy-backed acknowledgement schemes used in some of the newer communications line control disciplines such as IBM's SDLC [24] or the International Standards Organization's (ISO's) HDLC [25]. A somewhat different version of a piggy-backed acknowledgement scheme, designed specifically for the type of completely synchronous communication anticipated for the radar network is suggested here. The major difference between the type of communication in the radar network and that handled by such disciplines as SDLC or HDLC, with regard to potential acknowledgement schemes, is that no known percentage of the messages flowing in a network managed by SDLC or HDLC is expected to go in a particular direction, but the completely synchronous operation used by the radar network implies that return messages always come back at predefined times. Hence, the radar network is always guaranteed to send a return message, on which one or more acknowledgements can be piggy-backed, within a known period of time. Another significant difference is that SDLC and HDLC always assume that erroneous messages are repeated, while repetition may or may not be required with the radar network.

An example of a reasonable scheme for using piggybacked acknowledgements will be developed for a case where transmission is half-duplex, with the division of capacity between the two directions of transmission restricted to be 1/4 and 3/4, or 1/2 and 1/2, or 3/4 and 1/4, i.e., each end of the link could use either 1,2 or 3 of every 4 packet time slots for its transmissions with the other end getting the remaining slots. (Modifications for numerous other possible divisions of capacity between the two directions of transmission should be obvious once this case has been explained. Obvious simplifications can be made if true full

duplex operation should prove to be feasible.) The suggested approach yields either a positive or a negative acknowledgement for each message at the cost of six extra bits in each message (except in what should be extremely rare error recovery situations).

Let a, b, c, d, e and f represent the six bits used for acknowledgements. (They could occur at any desired location in the radar track report packet format discussed earlier, probably in successive bit positions.) Bits a, b and c will be discussed first, with discussion of d, e and f following. The format for information flow will be assumed to be 3 packets in one direction followed by 1 in the other, then a repetition of this pattern, for unbalanced data flows, and a simple alternation of one packet in one direction, then one packet in the other direction, for balanced flows.

Consider unbalanced flows first. The end receiving three packets for each one it transmits will use bits a, b and c for positive or negative acknowledgements of the three packets, with 1 representing a positive acknowledgement and 0 a negative acknowledgement. Thus, abc = 101 will positively acknowledge the first and third packets and negatively acknowledge the second (i.e., errors have been detected in the second packet only). The end receiving single packets will, similarly, use 1xx and 0xx for positive and negative acknowledgements, respectively. (An x in this notation indicates a "don't care" bit, whose value is immaterial.) There is some slight excess redundancy in acknowledgements sent from the end transmitting three packets in a row (and using the 1xx or 0xx formats) since such acknowledgements will be repeated three times, but the excess overhead is minimal. (Also, the repetitions can help to insure acknowledgements actually get through correctly. Since a reasonably powerful error detecting code is assumed to be used, the first of the three acknowledgements for which a valid error check is obtained might well be accepted; taking a majority vote among the three

repetitions should give minimal additional gains in insuring correct reception.)

For balanced flows, an analogous acknowledgement mechanism can be used, but with 1xx and 0xx formats used in both directions.

The formats suggested should, if the packets in which they are encoded are received correctly, give positive and negative acknowledgements for all packets. If errors occur in packets, though, additional complications come in since the locations of errors are unpredictable. Thus, at any time when a negative acknowledgement has been received for a particular packet, the acknowledgement bits accompanying the negatively acknowledged packet cannot be assumed to be correct. An example where acknowledgements would need to be repeated is the case above where an acknowledgement pattern is 101; in this case, the acknowledgement bits from the second message would need to be repeated even if the message itself were not critical enough to be repeated. (This step is necessary since messages included in the packets which were negatively acknowledged may or may not be repeated.)

Bits d, e and f are used for repetitions of acknowledgements bits in any cases where this is necessary. A brief itemization of possible cases (not included here) indicates that all possible cases where repetition of acknowledgement bits might be desirable can readily be handled in this manner, except for what should be extremely rare cases where a number of successive repetitions of the acknowledgement bits fail to get through. Special error recovery routines may be needed for these rare occasions.

SUMMARY

A description of the current status of plans for a future Air Force tactical radar network has been given and some of the major problem areas expected to be encountered in developing such a system have been mapped out. Major areas covered have included typical network topologies, projected forms for track reports, possible approaches to tracking, and initial communications protocol considerations. The final area listed, communications protocols, has been studied the most thoroughly of the areas listed, with some suggestions made concerning synchronous transmission schemes, major classes of data transmissions, error handling mechanisms, routing algorithms, making modifications to the network structure, and possible techniques for handling acknowledgements.

This report is only a brief introduction to a highly complex network which is under conceptual development. Hopefully, a number of the suggestions here will be useful in further definition of the network, either being incorporated into network design or stimulating the development of superior techniques. Full development of a suitable network will require a large-scale effort over a number of years. The author hopes that he will be able to contribute usefully to this effort by means of this report, by completion of additional studies being done under AFOSR Grant No. 77-3339, and through some continuation studies for which a proposal is currently being written.

ACKNOWLEDGMENTS

The author would like to express his appreciation to the personnel in the office of the Deputy for Development Plans, Headquarters Electronic Systems Division, who have assisted him in this study. Especial thanks go to D. Brick, for his assistance in getting the study underway and to O. Wech for his assistance throughout the study. Discussions with personnel at Mitre Corporation, at Rome Air Development Center and at MIT Lincoln Laboratories have also been helpful.

REFERENCES

1. Spragins, J. D., "Potential Technology Breakthroughs in Next Generation Radars," Participant's Final Report for 1976 USAF-ASEE Summer Faculty Research Program, 27 August, 1976.

2. Cartledge, L. and O'Donnell, R. M., "Description and Performance Evaluation of the Moving Target Detector," Report No. FAA-RD-76-190, Lincoln Laboratory, Lexington, Massachusetts, 8 March, 1977.

3. Electronic Systems Division, USAF, "Statement of Work for Data Systems Requirements Study Related to Future USAF Tactical Air Surveillance and Control Improvements," 1977.

4. Hurney, W. T., Et al., "Air Force Tactical Air Control System, Facilities and Functions," MITRE Corporation Technical Report MTR-3299, Oct. 1976.

5. Workman, B. J., "An Overview of the Joint Tactical Information Distribution System (JTIDS)," MITRE Corp. Technical Report MTR-3228, April 1976.

6. "System Specification for the Region Operations Control Center Segment of the Joint Surveillance System," Specification No. ESD-RS 968H, 1 June 1976.

7. O'Donnell, R. M., Lincoln Laboratories, Lexington, Mass., private communication.

8. Karp, P. M., "Origin, Development and Current Status of the ARPA Network," Seventh Annual IEEE Computer Soc. Int. Conf., Dig. Papers, pp. 49-52, 27-28 Feb., 1 March, 1973.

9. Peterson, W. W. and Weldon, E. J., Jr., Error Correcting Codes, 2nd ed., MIT Press, Cambridge, Mass., 1972.

10. Waldman, A., Kulpinski, R. J., Hunt, S. H. and Cook, C. E., "On-Site Processing Requirements for Tactical Netted Surveillance," MITRE Corp., Bedford Operations, Working Paper No. 20566, 20 Jan. 1976.

11. Advanced Planning Office (XRT), Headquarters Electronic Systems Division, USAF, Hanscom Air Force Base, Bedford, Mass., private communications.

12. Doll, D. R., "Multiplexing and Concentration," Proceedings of the IEEE, Vol. 60, No. 11, pp. 1313-1321, Nov. 1972.

13. Chu, W. W., "Demultiplexing Considerations for Statistical Multiplexors," IEEE Transactions on Communications, Vol. COM-20, No. 6, Part II, pp. 603-609, June 1972.

14. Gabbard, O. G. and Kaul, P., "Time-Division Multiple Access," IEEE Eascon Conference Record, pp. 179-184, 1974.

15. Scholtz, R. A., "The Spread Spectrum Concept," IEEE Transactions on Communications, Vol. COM-25, No. 8, pp. 748-755, Aug. 1977.

16. Burton, H. O. and Sullivan, D. D., "Errors and Error Control," Proceedings of the IEEE, Vol. 60, No. 11, pp. 1293-1301, Nov. 1972.

17. Wozencraft, J.M. and Jacobs, I.M., Principles of Communication Engineering, John Wiley and Sons, Inc., New York, 1965.

18. Schwartz, M., Computer-Communication Network Design and Analysis, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1977.

19. Gitman, I., "Comparison of Hop-by-Hop and End-to-End Acknowledgment Schemes in Computer Communication Networks," IEEE Transactions on Communications, Vol. COM-24, No. 11, pp. 1258-1262, No. 1976.

20. Bochmann, G. V., "Communication Protocols and Error-Recovery Procedures," Proc. ACM SIGCOMM/SIGOPS Interprocess Communication Workshop, Santa Monica, Calif., pp. 45-50, March 1975.

21. Zafiropulo, P., "A New Approach to Protocol Validation," Proc. International Conference on Communications, Vol. 2, pp. 259-263, Chicago, Ill., June 1977.

22. West, C. H., "An Automated Technique of Communications Protocol Validation," Proc. International Conference on Communications, Vol. 2, pp. 264-268, Chicago, Ill., June 1977.

23. Eisenbies, J. L., "Conventions for Digital Data Communication Link Design," IBM Systems J., Vol. 6, No. 4, pp. 269-302, 1967.

24. Donnan, R. A. and Kersey, J. R., "Synchronous Data Link Control: A Perspective," IBM Systems J., Vol. 13, No. 2, pp. 140-162, 1974.

25. Neumann, A. J., et al., "A Technical Guide to Computer-Communication Interface Standards," NBS Technical Note 843, Aug. 1974.