1988

(2)

A RAND NOTE

RJARS: RAND's Version of the Jamming Aircraft and Radar Simulation

William Sollfrey

207 pg

RAND

The research reported here was sponsored by the United States Air Force, contract No. F49620-91-C-0003; the United States Army, contract No. MDA903-91-C-0006; and by the Defense Advanced Research Projects Agency and the Director of Defense Research and Engineering, contract No. MDA903-90-C-0004. These federally funded research and development centers are sponsored by the U.S. Air Force, the U.S. Army, and by the Office of the Secretary of Defense and the Joint Chiefs of Staff.

Published 1991 by RAND
1700 Main Street, P.O. Box 2138, Santa Monica, CA 90407-2138

A RAND NOTE

N-2727-1-AF/A/DARPA/DR&E

RJARS: RAND's Version of the Jamming Aircraft and Radar Simulation

William Sollfrey

Prepared for the
United States Air Force
United States Army
Defense Advanced Research Projects Agency
Director, Defense Research and Engineering

RAND

PREFACE

RJARS is an engagement level model that simulates air-to-ground and ground-to-air combat, primarily the latter, treating the combatants as individuals rather than aggregating. It has been designed to consider terrain masking, multipath and clutter, and flight dynamics in order to more carefully evaluate jamming effectiveness and mission attrition.

The model is an extensive development of JARS (Jamming Aircraft and Radar Simulation), which was originally developed at the Johns Hopkins University Applied Physics Laboratory. The redevelopment effort began in 1989 for studies of electronic combat. Funding for the multiyear development effort was provided by a coordinated set of RAND sponsors that included the Air Force Assistant Deputy Chief of Staff for Plans and the Assistant Deputy Chief of Staff for Operations, the Deputy Director of Defense Research and Engineering (Tactical Warfare Programs), the Defense Advanced Research Projects Agency, and the Army's Assistant Deputy Chief of Staff for Force Development. This effort also drew on exploratory research funds from RAND's Project AIR FORCE, the Arroyo Center, and the National Defense Research Institute, federally funded research and development centers (FFRDCs) sponsored, respectively, by the U.S. Air Force, by the U.S. Army, and by the Office of the Secretary of Defense and the Joint Chiefs of Staff. The work was conducted through a series of related projects that included "Electronic Combat in Support of Defense Suppression Operations" (Project AIR FORCE), "Concept Analysis Environment" (National Defense Research Institute), "Future of Army Aviation" (Arroyo Center), and the "Joint Close Support Project," which is supported by all three FFRDCs.

The current version of RJARS considers sorties in which aircraft carrying warning receivers, jammers, anti-radiation missiles, and air-launched cruise missiles fly against a defensive system with search, acquisition, and tracking radars, IR and optical systems, surface-to-air missiles, artillery, and a command, control, and communications system. All equipment parameters and scenarios can be varied. The program is fast, efficient, and compact. RJARS can operate independently or in conjunction with other simulations that have been developed at RAND. In particular, coordination with RAND's CAGIS (Cartographic Analysis and Geographic Information System) program enables the inclusion of detailed terrain parameters, coordination with the flight planners BLUE MAX (for fixed-wing aircraft) and CHAMP (rotary wing aircraft) provides flight paths over the terrain including aircraft dynamics, and coordination with JANUS allows treatment of air effects on ground combat.

This Note is an update of N-2727-AF, December 1988. It should be of interest to analysts and mission planners who wish to treat air attack versus ground defense systems from either viewpoint, to ascertain the survivability of missions against various defenses, to evaluate the effectiveness of ground equipments in either the unjammed or jammed condition, the effects of radar cross section, the significance of cutting communications, the results of changing any of the equipment or scenario parameters, or almost any other problem which may arise in the treatment of such systems.

This Note describes RJARS as it was during September 1990. It is expected that there will be improvements and additions to the program. Any reader who wishes further information should contact Dr. William Sollfrey, RAND, 1700 Main Street, Santa Monica, CA 90407-2138, Telephone 213-393-0411, Extension 7222.

SUMMARY

RJARS (RAND's version of the Jamming Aircraft and Radar Simulation) is a many-on-many computer simulation involving aircraft, radars, IR and optical systems, jamming systems, offensive and defensive missiles, and a command, control, and communications system for the defense. The simulation can handle hundreds of aircraft and radars. Terrain masking, clutter, and multipath are included. It is an extensive development by RAND of the computer program JARS (Jamming Aircraft and Radar Simulation), which originated at the Johns Hopkins University Applied Physics Laboratory (Refs. 1, 2). RJARS has been designed to treat sortie operations and evaluate jamming effectiveness and mission attrition at a level of detail that includes reasonable refinements of equipment operation without excessive calculational complexity.

At RAND, RJARS operates in conjunction with the Army's JANUS ground combat model, the CAGIS (Cartographic Analysis and Geographic Information System) terrain model, and several flight planners. All operations of RJARS have been programmed both for independent operation and for use of these external programs. A Revision Control System (RCS) keeps RJARS up to date and consistent for all users.

The parameters of all equipments (aircraft, receivers, jammers, airborne missiles, radars, and surface-to-air missiles (SAMs)) are stored in library files that are updated and maintained off-line. A simulation is run under the control of a scenario. Among the scenario inputs for the offense are the number and types of the aircraft, the equipments carried on board, and their flight paths. Flight paths can be described in terms of specified commands, or may follow the output of an off-line flight path generator. Aircraft can turn, climb, and accelerate, turn jammers on and off, enter or exit formation flying, and launch weapons. The defensive scenario includes the positions and types of all radars and missiles and who reports to whom. All radars and SAM launchers are at fixed positions. Infrared (IR) and optically aimed missiles and anti-aircraft artillery are included with a separate command structure in which they receive cueing data from the communications system.

The operation of the program may be understood by following a simple scenario. Initially, only the defensive long-range search radars and the searching optical systems are activated. Suppose an aircraft (the term aircraft includes both fixed-wing and helicopters) comes into the field of view of some radar. A probabilistic detection will be performed, which may take several scans to establish identification. When detection is confirmed, the radar will transmit this information to its "site". If the communications channel is working, the site will report the new detection to its "command site". At the command site, the information will be evaluated and, if possible, a tracker will be assigned from among those which report to that specific command site. (The terms "tracker" and "tracking radar" are used synonymously throughout this Note.) There are a number of conditions which must be satisfied; for instance, if a particular tracker is to be assigned the projected path of the aircraft must come within the maximum operating range of the SAM associated with that tracker.

When a tracker is assigned, its associated acquisition radar is turned on after a delay determined by the communications delays (uplink and downlink) and the decision delay at the command site. The acquisition radar performs a probabilistic detection like that of the search radar, and if successful turns the tracker on. RJARS radars are multifunctional, so one equipment may perform search, acquisition, and tracking functions. The tracker then will follow the target, with errors determined by target glint, signal-to-noise ratio, and jamming effects. If the communications system had not been operating (the scenario provides for cutting or connecting links at specific times), the site will perform the assignment task autonomously, using only the equipment that it controls directly.

Infrared and optically aimed weapons are controlled differently. They search specified sectors using algorithms for detection and recognition of targets against sky or ground background. If a target is detected and recognized, either the IR missile waits for lock-on, is counted down and launches, or the gun is aimed and fires. If the optical signal is lost after recognition, the equipment returns to search. If any command system has detected a target, the warning information is broadcast throughout the field. If the target is potentially in range of a searching optical system, the search sector is narrowed for improved detection.

While these defensive operations are proceeding, the aircraft will continue their flight maneuvers. They turn to the appropriate headings, climb, descend, pitch, bank, or accelerate, and launch weapons when commanded by the scenario. The warning receivers detect the radars and catalog the information. The jammers may be directed to jam radars of any or all classes (search, acquisition, or track). Jammers can employ noise jamming, any combination of range or angle deception, or one of several varieties of towed decoys, depending on the capability of the jamming equipment and the choice of jamming technique designated by the library as appropriate for that radar. Infrared systems may be decoyed by flares. Rudimentary IR missile warning systems are on the aircraft, and rudimentary flare rejection systems are on the IR missiles. Jamming of infrared or optical systems is not included in RJARS at this time.

Deception is simulated by matching the jammer bandwidth to the radar bandwidth. For search radars, the effect on the probability of detection is ascertained. For tracking radars, the errors in range and angle are determined, and the signal-to-noise ratio is compared to the breaklock level (deterministic rather than probabilistic comparison). If the signal to noise ratio is below the break-lock level in range or angle, the tracker either cannot establish lock on the target or loses simulated lock if established, and cannot guide a SAM if one has been launched. There is no attempt to simulate the detailed operation of range or angle tracking circuits. If a tracker is jammed while the target aircraft is approaching, it will attempt to lock on to the aircraft for a time specified in the radar library. If it is jammed while the target is receding, it will drop track.

If a tracker establishes lock on its target for sufficient time, it will count down and launch a SAM at the target. Countdown times are found from a probability distribution. Launch success is calculated by a random draw against the launch reliability. If the launch is successful, the SAM will fly on a trajectory in which its acceleration is determined by a thrust and drag program, and its steering is either command-guided toward a predicted point of intercept or uses proportional navigation with a semiactive radar or IR system. The SAM aerodynamics are represented by a first-order lag in the response to the guidance command. The command data are corrupted by the radar errors. When the SAM reaches its closest approach to the oriented target, the miss distance is calculated by adding a randomly oriented normally distributed guidance CEP to the radar-produced error, then calculating the kill probability as a piecewise linear function of the miss distance. If there is an actual hit, the kill probability is unity. Otherwise a random draw against the kill probability determines the success. If the aircraft has been shot down by the SAM, it is removed from the simulation. If not, the tracker begins a second countdown (or more if necessary).

Data have been collected on the effectiveness of jammers against various SAM systems. These data may be used to determine the reduction in kill probability of a SAM, rather than the detailed jamming calculations. Data are available both averaged over all cases and as a function of signal-to-noise ratio.

If the aircraft are carrying anti-radiation missiles (ARMs), they may be launched at the trackers. Frequently there is a close race, with the ARM being launched first but with slower speed than the SAM. If the ARM kills the tracker (another probabilistic calculation) before the SAM arrives at the aircraft, the SAM will lose guidance and continue ballistically. Then if there is no interception following a specified time interval, the SAM will self-destruct.

If the tracker goes off the air, the ARM will continue its flight with reduced kill probability. The alternate condition in which it diverts to another target was not implemented, a limitation of the simulation. Air-launched cruise missiles are treated as additional aircraft with their own scenario parameters.[1] The interplay of aircraft, tracker, and missiles continues until the aircraft is shot down or escapes from the region.

Terrain may be included in the calculation in several ways. There may be no terrain, with all calculations performed over a smooth spherical earth. Terrain may be included directly, with the elevation of the line of sight over the terrain between each aircraft-radar pair calculated at each time step and the result used to determine instantaneous visibility. If the simulation is lengthy or repetitive operation is to be employed, a preprocessor mode may be used to calculate the visibility intervals for each pair. These visibility intervals are then used to control the main simulation.

Multipath and clutter effects are included in the radar calculations. The ground defenses include not only radars but infrared missiles and optically aimed guns. Background effects on such devices are included. Terrain effects may be calculated using CAGIS, which is more efficient than RJARS in finding visibilities between radars and aircraft or ground clutter points. Also, the ridges along various directions from each ground equipment are calculated, preferably by CAGIS, to determine if the ground is visible to the radar or if the optical system sees the target against a sky or a terrain background.

The sequence of operations as described will proceed until either the end of the simulation time is reached or all aircraft have been shot down. During the run, at scenario-selected time intervals, information is printed out on the aircraft positions and maneuvers, the search or acquisition radars' current observations, the tracker measurements and errors, the ARM's position and destination if launched, and the SAM's position if launched. When the SAM is closing on its target (less than 1000 feet) the details of the SAM trajectory are printed. Events such as a detection, launch, or kill are printed as they occur.

---

[1] RJARS addresses only cruise missile survivability, not effectiveness.

At the end of the simulation, summaries are printed showing all field of view entries, detections, and exits for each search radar and aircraft pair; times of assignment, acquisition, and tracking duration for each tracking operation; launching and result for each ARM; and launch and intercept times and results for each SAM. Box scores show how the aircraft fared in accomplishing their missions (reaching destination, surviving ingress and egress), and how the SAMs performed their defensive mission. Statistics on killing intervals and kill probability density may be collected. When RJARS is operated in conjunction with the JANUS ground combat model, these statistics are transmitted to JANUS for its use in determining the effects of coordinated air strikes on ground combat operations

RJARS may be operated in Monte Carlo repetitive style. The number of repetitions is specified in the scenario. At the end of each run, all the variables are reset to their initial values except the random number generator, which is allowed to continue from its present value. The resulting statistical information must be processed offline after completion of the Monte Carlo sequence, except for the data transmitted to JANUS. When operating Monte Carlo, the details of the simulation are printed only for the first run. On the ensuing runs, only the status events (detections, launches, etc.) and the summaries are printed.

RJARS now includes a graphics calculation. This is provided as a file in CAGIS format, and is the only nonportable part of RJARS. With the graphics, the terrain and defenses may be depicted on the screen, along with the aircraft, SAM flight paths, and their interactions. As an aircraft progresses along its path, it is initially in black. The path color changes to gray when a searcher detects the aircraft, to blue when a tracker is in acquisition, to yellow when a tracker is in track, and to orange when a SAM or shell has been launched. SAM paths are in red. An endgame miss is a small red star, an endgame kill is a large red star. Icons are used for the defenses to indicate the type of radar, SAM, or gun. Targets for the aircraft may also be shown. Iterations may be presented in sequence. The paths may be laid down all at once, or the flights may be time-stepped. The latter is usually much more descriptive. The graphics operation is an RJARS postprocessor, for which the data file is prepared during the RJARS run, and the graphics program may be run at any time.

The variables treated stochastically are as follows:

- Defense system element reliability. A uniformly distibuted random draw against a reliability which is a property of the type of element.
- Radar frequency. A random value uniformly distributed over the frequency range available to the radar is selected.
- Radar initial azimuth. A random variable uniformly distributed from 0 to 360 degrees is selected.
- Search radar probability of detection. This is calculated by a formula relating probability of detection to signal-to-noise ratio, then compared to a random variable uniformly distributed between zero and one. If the probability of detection exceeds the value of the random variable, a detection takes place.
- Search radar errors. The rms errors in range, azimuth, and elevation are calculated by formulas, then multiplied by independent random variables normally distributed with zero mean and unit variance.
- Tracking radar errors. Glint and noise (including jamming) are treated independently. The rms errors in range, azimuth, and elevation are calculated, then the current errors are determined using correlated random variables, distributed according to three independent two-variable normal distributions with zero mean, unit variance, and correlation between the variables (say the present and previous values of range error), which depends on the glint frequency or servo bandwidth.
- Anti-radiation missile kill probability. The library value of the kill probability of an ARM of the appropriate type against a radar of the appropriate type, reduced by a time-dependent factor if the radar goes off the air, is compared to a random

variable uniformly distributed between zero and one. The library value is actually the product of the launch reliability and kill probability. If the calculated value exceeds the random value, a kill takes place.

- Surface-to-air launch reliability. The library value is compared to a random variable uniformly distributed between zero and one. If the library value exceeds the random value, a launch occurs.
- Surface-to-air missile CEP. This is treated as a vector, whose magnitude is normally distributed with the proper rms value, and whose direction is uniformly distributed over the unit sphere.
- Surface-to-air missile kill probability. This is calculated as a function of miss distance, then compared to a random variable uniformly distributed between zero and one. If the kill probability exceeds the random value, a kill takes place.
- Random phases for multipath or clutter calculations.

The radar operating frequency and initial azimuth are calculated at the beginning of the program and retained thereafter. The other random variables are each calculated at the appropriate time (search radar variables at each time step, tracking radar variables at each time step when the tracker is on and each subdivided time step when a SAM has been launched, reliability at launch, and CEP and kill probabilities at the time of kill).

RJARS operates on UNIX systems in the C language. Dynamic allocation permits RJARS to use the smallest amount of memory space compatible with the size of the scenario.

A more detailed description of the sequence of operations is presented in Sec. II. An analytical section (Sec. III) provides the theoretical basis for the program. A user's guide (Sec. IV) shows how to prepare input files and operate RJARS. A programmer's guide (Sec. V) presents programming details and a glossary of the approximately 1100 variables used in RJARS. Program flow charts appear in the Appendix.

An implicit assumption in RJARS is that the laydown is truly known, implying perfect intelligence. It is also assumed that the offensive and defensive equipments are working perfectly, ignoring problems of electromagnetic compatibility and electromagnetic interference other than jamming. These problems may be considered in the future. RJARS should be regarded as a simulation, not as a true representation of the real world.

Further developments of RJARS are planned during the upcoming year. The input data will be converted to a menu-driven operation that should be more user-friendly. A statistics package will be added. The radar modelling will be improved to provide better simulation for CW and pulse doppler radars. Multipath, clutter, and electro-optical (EO) backups will be investigated. The command and control will be modified to provide a three-level structure with intercommunications, skip echeloning, and other variations. The aircraft vulnerability treatment will be expanded to include better dependence of kill probability on aircraft and missile type and a probability of kill given a hit model. Target priority and firing doctrine will be investigated. It is expected that this work will be funded from Project AIR FORCE and the Arroyo Center.

ACKNOWLEDGMENTS

It is a pleasure to thank program directors Natalie Crawford and Bruce Don and project leader Fred Frostic for support in developing this improved version of RJARS. Judy Lender established and maintained the configuration control system that keeps RJARS the same for all users. Jack Ellis performed miracles of data collection and interpretation. Keith Smith developed the coupling of probability effects between RJARS and JANUS. Jim Gillogly originated the method of dynamically allocating variables. The graphics were all developed by Gail Halverson. Al Zobrist, the chief modeler and developer of CAGIS, performed yeoman service in the coordination between CAGIS and RJARS. James Jennings and Sally LaForge, respectively the adaptor of BLUE MAX and the developer of CHAMP, labored to make their programs work in conjunction with CAGIS and RJARS. The numerous users, especially Bill Dean, Ted Harshberger, and Jerry Stiles, found innumerable bugs before they became seriously contagious. The comments of reviewer John Clark are much appreciated. Anybody else who helped the author is hereby thanked anonymously.

CONTENTS

FIGURES

I. INTRODUCTION

The original version of the Jamming Aircraft and Radar Simulation (JARS) was developed at the Johns Hopkins University Applied Physics Laboratory (Refs. 1 and 2). Its capability is well described by the following quotation (Ref. 1, page 1-1):

"The Jamming Aircraft and Radar Simulation (JARS) is a PL/I computer program which simulates a many-on-many scenario involving support jammers, strike aircraft, and early warning radars that are netted to a user-defense system. The program provides the user with the opportunity to evaluate jamming techniques and tactics against the radar defense system. A probabilistic determination of target detection is obtained for each radar and each radar site against each aircraft."

Essentially the original version of JARS determines how aircraft carrying jammers will interact with search radars. The aircraft fly prescribed paths, and jammers are turned on and off under user control. Only search radars are included, and the aircraft and radars are immortal (no weapon-type interactions). The only significant outputs are the times in which aircraft are in detection and the times that radars are jammed. JARS is an excellent program which performs rather limited objectives.

In the course of a study on electronic warfare, it was decided at RAND that we would upgrade JARS to a full-fledged sortie simulation. The resulting program, named RJARS (RAND's version of the Jamming Aircraft and Radar Simulation) is several times as long as the Johns Hopkins version. It enables us to investigate a much greater variety of offensive and defensive configurations and determine the effectiveness of jamming techniques, including noise and simulated deception, when the offense and defense are interacting.

RJARS can operate independently or in conjunction with other programs. It has been translated into the C programming language to work under the UNIX operating system on the Sun work stations in the RAND Military Operations Simulation Facility (MOSF). As such, RJARS works with the Army's JANUS ground combat model to provide input threat laydowns, and RAND's CAGIS (Cartographic Analysis and Geographic Information System) terrain mapping model to provide terrain and other inputs. The flight path generators BLUE MAX (fixed-wing) and CHAMP (helicopters) associated with CAGIS can be used to prepare flight paths for RJARS. Detailed procedures in CAGIS may be used to prepare the RJARS scenario and other input simulation files via automatic data preparation instead of manual calculation. The graphics output of RJARS is written to be read by CAGIS. While all of these have been very valuable to RAND's use of RJARS, an external user, who lacks these associated programs, can still use RJARS by itself for everything but the graphics. The interaction among the computer programs is depicted in Fig. 1.

The PL-I version of RJARS described in the first edition of this report [Ref. 3] was updated to include a few of the phenomena that were added to the C version in the period between June 1, 1988 (when the original N-2727-AF was actually completed) and the present. It is no longer being kept up to date. Only the C version will be treated hereafter.

The C programming language requires that the names of all variables be fully qualified. This would lead to very complicated description; hence, we have retained the PL-I type notation for the variables in the text. For example, the latitude of aircraft J is designated ACLAT(J), rather than the full C name AIRCRAFT[J].AC.LAT. The glossary at the end of the programmer's section lists all variables in the PL-I notation, with structures represented in the first two or three letters of the variable name.

The aircraft now carry warning receivers, jammers under warning receiver control, towed decoys, air-launched cruise missiles,[1] and anti-radiation missiles. The ground-based defensive system includes search, acquisition, tracking, and illumination radars, IR and optical

---

[1] Air-launched cruise missiles are treated as additional aircraft with their own scenario parameters. It is their survivability that is being assessed, not their effectiveness.

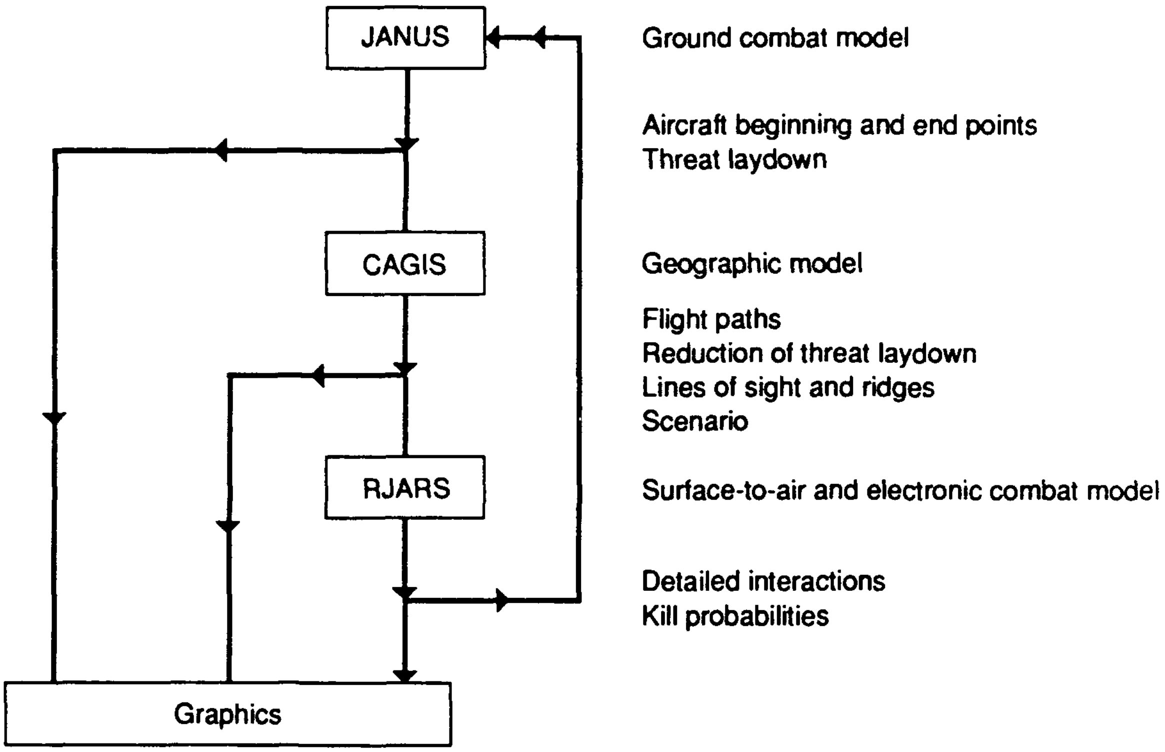


Fig. 1—Connections among computer programs to form land-air combat model

systems, surface-to-air missiles, anti-aircraft artillery, and a command, control, and communications system. Terrain is used to determine masking, clutter, and multipath effects. The entire program may be operated in Monte Carlo fashion. Input data are contained in two types of files--library files with the properties of specific equipments, and scenario files for particular runs.

RJARS may be employed to investigate a great variety of problems that require an intermediate level of detail. Some potential applications are the following:

1. Medium-scale attrition studies. Flights of aircraft carrying jammers and anti-radiation missiles (ARMs) may be operated against surface-to-air missile (SAM) and anti-aircraft artillery defenses, and the resulting attrition and mission success ascertained. RJARS employs dynamic allocation for its variables, and its problem size and speed limits are set by the properties and load of the machine, not by any inherent limitations in the program. Such attrition studies would be of value to mission planners.
2. Jamming effectiveness studies. Aircraft may be flown with different jammers, and the corresponding mission successes compared. The effectiveness of jamming techniques can be investigated. The absolute and relative effectiveness of various jammers would be valuable information to both mission planners and equipment designers. This assumes that the response of the several types of radars to jamming techniques is known. The real world may be very different from the world of the simulation.
3. Effects produced by terrain, of primary interest for low-flying aircraft missions. The delays and interruptions in radar coverage produced by terrain masking, and the resultant changes in defensive capability, can be evaluated to determine appropriate routes for attack missions. The effects of clutter or multipath on the radar configuration can be determined.

4. Radar cross-section effects. The improvement in mission success--of the aircraft reaching target or surviving the mission--produced by reducing the radar cross-section of the aircraft can be studied.
5. Communications cutting effects. RJARS organizes the defensive system so each radar and each SAM launcher reports to a "site", and each "site" reports to a "command site". Communications proceed upward from a site to its command site, describing the detection of aircraft, and downward from a command site to any of its sites, describing tracker assignment. There are delays on each link, and a decision delay at the command site. A typical configuration is shown in Fig. 2 (p. 12). When communications links are cut, RJARS sites revert to autonomous operation. Each isolated site (one whose communications have been cut) assigns trackers that report to it as if there were no other defenses present. The overall defensive assignments are changed, the system delays are reduced, and the resource allocations will be different. The resulting changes in mission success can be studied to ascertain the value of such communications cutting.
6. Resource allocation and defensive saturation studies. A large attack can saturate the defenses, since the number of trackers is limited (RJARS does not include track-while-scan radars), and thereby produce mission successes that cannot be deduced from one-on-one studies. Also, the limited number of missiles at a launch site can affect the defensive capability, since sites can run out of missiles. Mission planners can use RJARS for such studies.

These are but a few possible applications of RJARS. The interested reader can undoubtedly think of ways to apply it to his or her particular field of interest.

RJARS has limitations that prevent the study of various problems. Among these are:

1. All aircraft maneuvers have to be preloaded in the scenario. Reactive maneuvers, such as attempts to evade attacking missiles, are not included. However, reactive maneuvers can be treated in an equivalent fashion by running a scenario, ascertaining which defenses interact with the attack, and then modifying the flight paths externally. The use of flight path generators facilitates this process.
2. The airborne warning receivers do not apply priority tables to the detected radar signals, so all jammed radars in a frequency band of the jammer receive equal allocations of jammer power. Furthermore, the warning receivers are assumed to detect the signals as if they were radiated independently. Problems of interleaved pulse trains and other "high signal density" effects are not considered.
3. The command and control system for the defense is limited to site and command site up and down interactions, with no cross connections between command sites. This prevents consideration of large-scale command structures.
4. Communications operate in an on-off fashion, so studies of effects of communications jamming are limited to circumstances in which jamming is either completely effective or totally ineffective.
5. Rain attenuation, which can be significant at the higher radar frequencies, is not modeled in RJARS, so the simulation corresponds to frequencies below 10 GHz, or clear weather conditions for frequencies above 10 GHz. However, IR and optical attenuation are included.
6. The phenomena associated with ducting propagation are not included. They can cause significant effects on radar propagation over water paths, which could be important if RJARS were extended to treat naval surface-to-air operations.

7. As mentioned in the Summary, problems of electromagnetic compatibility and electromagnetic interference are not included. These may play a very important role in determining if the equipments are actually working in the manner in which they are being simulated.

It is expected that there will be improvements and additions to RJARS during the coming year. The treatment of CW radars will be improved to consider how they handle information differently from pulse radars. Clutter for CW seekers will have an improved algorithm for calculating the area from which clutter is received, since the present algorithm is both inaccurate and wasteful of computing time. Phased array radars will be included. Their problems include sector coverage, raster or random scan patterns, dwell time phenomena, and track-while-scan operation. Tracking of multiple targets by a single radar, use of multiple illuminators, and simultaneous control of several SAMs by a single radar will be incorporated. The theory of multipath over curved irregular ground will be improved. Electro-optical backup for the radars will be incorporated. The gun model will be improved, and the endgame calculation will include allowance for the probability of kill given a hit for the various aircraft and weapon types. The command and control structure will be revamped to provide three-level structure, skip echeloning, matching operating modes to the tactical situation, giving the user a choice of command and control models for the several SAM types, and a general improvement in the communications model. Not all of these operations may be accomplished.

We first describe the simulation to depict the sequence of decisions. Next, an analytical section provides the theoretical basis for the program. A user's guide shows how to prepare input files and operate RJARS, and a programmer's guide presents programming details and a glossary of the approximately 1100 variables used in RJARS.

The text that follows is almost entirely mathematical or verbal. The author does not think in terms of flow charts, and has not used them while building the program. However, following a reviewer's suggestion, a set of flow charts has been included as an Appendix. The reader can refer to the charts while reading the text.

II. GENERAL PROGRAM DESCRIPTION

RJARS is a many-on-many computer simulation involving aircraft, radars, jamming systems, offensive and defensive missiles, and overall control. Hundreds of aircraft and radars may be included. It is implemented either in earth-based or internally referenced coordinates, exactly as in JARSM (Ref. 2). Like all simulation programs, it begins by reading the input data. These data are contained in two types of files--library files containing equipment parameters, and simulation files with the information for a particular run. There are nine library files, as follows:

1.	ACLIB	Aircraft performance data
2.	ACRCS	Aircraft cross-section data
3.	ARLIB	Anti-radiation missile data
4.	IRLIB	Infrared and optical data
5.	JMLIB	Jammer data
6.	MCLUT	Terrain properties
7.	RDLIB	Radar data
8.	SMLIB	Surface-to-air missile data
9.	WRLIB	Warning receiver data

and nine simulation files:

1.	SCENA	Simulation run parameters
2.	ACVIS	Aircraft visibility over terrain (input)
3.	ACSGT	Aircraft visibility over terrain (output)
4.	TERRA	Terrain heights
5.	BLUMX	Flight paths from BLUE MAX (fixed-wing)
6.	CHAMP	Flight paths from CHAMP (helicopters)
7.	RIDGE	Ridges as seen by radars (input)
8.	RDRDG	Ridges as seen by radars (output)

9. DISPL	Graphics display file (CAGIS format)

These files must all be prepared before a run is executed. (ACSGT, RDRDG, and DISPL are produced during the simulation.) Preparation details are given in the user's guide section. The library files may be maintained and updated as information is acquired on additional equipments. The scenario file (SCENA) is modified for each run.

Terrain may be omitted from the run or included in several ways. The terrain data may be read in and the visibility between aircraft-radar pairs calculated during a simulation. Alternatively, a preprocessor mode may be used, which traces the paths of the aircraft as specified in the scenario, and determines the time intervals for which each aircraft may be visible to each radar. The preprocessor output, ACSGT, may then be moved to the simulation input file ACVIS, which then provides visibility intervals for the full simulation. ACVIS may also be produced by the user from direct reading of maps. The visibility intervals as determined by the terrain are then used to control the simulation. If a small terrain region or a relatively small number of aircraft and radars are treated in the simulation, and if Monte Carlo operation is not involved, then it is usually best to incorporate the terrain effects directly in the simulation. For any large problem, or if Monte Carlo is included, the two-stage preprocessor simulator procedure will almost always be more efficient, since the terrain-masking calculation requires considerable computing time, and it is most desirable to not have to repeat the lengthy process.

Multipath and clutter effects involve determining whether the ground from which return is expected is actually visible to the radar. This calculation involves determining ridge and reappearance ranges from the radar to the terrain in various directions. This is also conveniently performed by a preprocessor, since the ridges remain the same throughout the simulation. When RJARS is operated in this mode, the preprocessor output is placed in a file RDRDG, which is then stored under an appropriate name and moved to the working file RIDGE when the configuration is to be treated by the full simulation. The RIDGE file may be very long if the terrain is complex and there are many radars, but this space consideration is overwhelmed by the advantage in running time for a simulation with many Monte Carlo iterations.

The files RIDGE and ACVIS are prepared by CAGIS in the RAND Military Operations Simulation Facility. Since CAGIS is a much more effective program for treating terrain than is RJARS, this greatly increases the overall efficiency of the combined systems.

Each aircraft carries a warning receiver, a jammer, and a scenario-selected number of air-launched cruise missiles and/or anti-radiation missiles. The aircraft are organized into groups, corresponding to formation flying. If an acquisition or track radar locks onto a member of the group, all the aircraft in the group will jam the radar. Aircraft may split from or join groups under scenario control.

The defensive system is organized into what we call for lack of a better name a two-level parallel system, consisting of sites and command sites. Figure 2 depicts a typical configuration A cluster of radars forms a site. All types of radars (search, acquisition, and tracking, the last with its associated SAM launcher) may be represented at a site. A site corresponds to the lowest level of field operations.

Each site is connected via communication links to a command site. The communications delays on each link are in the scenario file. These links may be cut or reconnected. Each command site will perform the assignment of all the defensive resources with which it is linked. When the link from a site to a command site is cut, the site reverts to autonomous operation and assigns its own resources. There is no interaction between command sites. Optically aimed weapons, such as shoulder-fired IR SAMs and free-standing guns, are not included in the command structure, but instead receive their information from communications broadcasting.

The program reads from library and simulation files until all data have been entered. If there are any reading errors, which cause an end-of-file condition to be reached, the program aborts and prints the appropriate error message. If no errors have occurred, the simulation begins, with the aircraft at their initial positions and vector

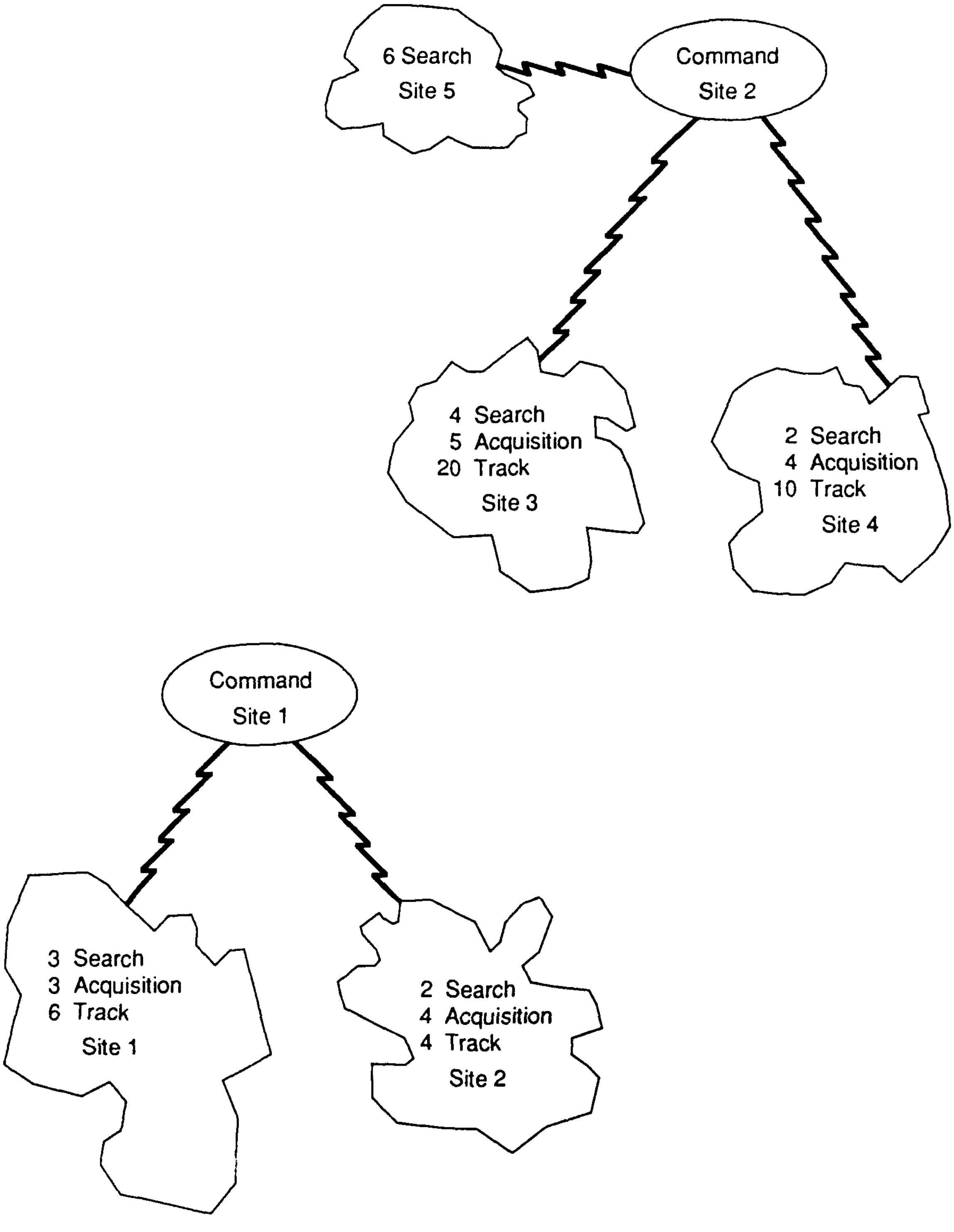


Fig. 2—Defensive configuration

velocities, and only the search radars and searching optical systems activated.

RJARS radars can perform a variety of functions. They may be purely search radars, or may be trackers which can search, acquire, track, and perhaps illuminate the target. Also, optically guided weapons employ simulated visual observers to perform their search and acquisition functions. The changing functions of the individual equipments are followed during the simulation.

The operation may be conveniently followed by considering a single aircraft. There is a specified field of action for the simulation, described by maximum and minimum latitudes and longitudes measured with respect to a coordinate origin. The aircraft will fly over the terrain and enter this field (initially the aircraft may be either inside or outside the field, but must be inside the terrain boundaries).

Equipments may be connected to the command and control system, or they may be autonomous. Shoulder-fired SAMs and free-standing guns are in the second category. Initially, all pure search radars and all optical autonomous systems are in search mode (the variable RDSTATE is L for the search radars, W for the optical systems). Also, the command system may contain command sites which have no pure search radar at any of their associated sites. For such command sites, the multifunction radar trackers are in the W state. Trackers at other sites are turned off (STATE 0).

The search radars and optical devices will be scanning, and eventually the aircraft will clear the terrain and become visible to some search radar. The warning receiver at the aircraft will detect the radar, if possible, and turn on the jammer if the scenario specifies that this aircraft should jam search radars. The jamming technique may be either noise or deception, depending on the characteristics of the radar and the capabilities of the jammer. This jamming operation will be deferred until the receiver has an indication that the radar has detected the aircraft, thereby avoiding acting as a beacon. Jamming in RJARS is limited to radars. No jamming is provided for IR or optical systems, but flares can be launched against the IR SAMs.

The search radar will receive the reflected signal from the aircraft, and if the suitably processed signal exceeds threshold for two of three successive scans, it will notify the site that it has a detection. This condition, which was also used in JARSM, should be sufficient to establish target identity. The radar will continue to scan, and the range, bearing, and elevation of the aircraft will be determined with errors that depend on aircraft glint and receiver signal-to-noise ratio as modified by jamming. Each search radar is equipped with a far sidelobe canceller, a form of automatic gain control that sets the system threshold so it is not triggered by signals, usually jamming, which come from directions in the far sidelobes of the radar antenna. All detections are probabilistic, with a detection probability that is a function of signal-to-noise ratio. Radar antenna patterns correspond to elliptical uniformly illuminated dishes, with stacked beams for search radars.

If the radar has information about a particular aircraft and the radar's site is connected to its command site, the information will be transmitted to the command site with a delay determined by the communication link. At the command site, the information will be catalogued and the command site will ascertain whether a tracker has already been assigned to this aircraft. (The terms "tracker" and "tracking radar" are synonyms in this Note.) If one has, nothing further need be done. If not, the command site examines the data on the aircraft from each radar that has seen it, and selects the position information that has the least error. It then looks among its trackers to choose the assignment. Conditions for assignment are:

1. The tracker and its acquisition radar must be alive and unassigned. Since either the tracker or the acquisition radar may be killed by an ARM, and an acquisition radar may serve several trackers, it is necessary to test that both are operating.

2. The aircraft must be visible to both the tracker and acquisition radar.
3. The aircraft altitude must be above the minimum and below the maximum altitude of operation for the SAM type associated with the tracker.
4. The aircraft must be farther away from the tracker than half the maximum range of the associated SAM. This is to provide enough time that the target may be acquired and tracked and the SAM counted down before the aircraft comes closer to the tracker than the minimum operating range of the SAM. The condition is too stringent for long-range SAMs.
5. The projected flight path of the aircraft must be inside or must pass into the circle around the tracker with radius equal to the maximum range of the SAM.
6. It is assumed that the SAM is launched at the earliest time possible (including the several system delays), and the interception point is calculated. This point must be within the maximum range of the SAM.
7. Trackers in RJARS have an assigned sector of responsibility. For most types, especially older vintage, the sector is a full circle. Certain types, including all IR SAMs and guns, have limited sectors. The target must be within the responsibility sector or be on a heading such that it will enter the sector.
8. Among the possibly several trackers that meet the first seven conditions, that tracker closest to the present position of the aircraft is assigned. If no tracker meets the conditions, the program will announce that fact.

When a tracker is assigned, its acquisition radar is turned on at a future time determined by the sum of the decision delay in the command site and the communications delay in the downlink between the command site and the site. For a multifunction radar, the tracker and acquisition radar are the same equipment, and the change is indicated by setting the variable RDSTATE to 'A'. If the assigned equipment is an IR SAM launcher (certain types of IR systems are under centralized control rather than autonomous) it goes directly into the track state--RDSTATE = 'Q'--and begins seeking the target.

If the communications link between a site and its command site is cut, the assignment operation proceeds as above, but only those search, acquisition, and track radars associated with the site may participate. The decision and up-and-down communications delays disappear.

The autonomous IR SAMs and guns receive their information in a different manner. Initially, they are in the W or searching state, scanning across their sector of responsibility using eyeballs, perhaps aided by binoculars. An algorithm gives the detection probability in terms of the optical contrast at the viewer. If one detects, it goes immediately to the A state, in which it uses the algorithm with a stronger criterion (recognition rather than simple detection) and instead of scanning stares in the apparent direction of the target. If it recognizes the target, it goes into track. Meanwhile, the command sites, which have been observing the battlefield, will broadcast the information on target position for each radar-detected aircraft. This broadcast information is received at all sites and used to cue the optical searchers to a narrower sector, so they have a better detection probability. The probability of receiving correct information at a site is at the moment set to unity. This was done because a survey of existing HF and UHF radio equipments on the possible sides of the simulated combat indicated that the range of successful transmission of information usually exceeded the dimensions of the RJARS combat field. It is expected that radio propagation will be studied to provide message probability for possible future versions of RJARS. The broadcast from each command site continues as long as that command site has information on that aircraft, but a site need be cued only once.

When the acquisition radar is turned on, it behaves like a search radar, but only collects information on its assigned target. Those aircraft in the group associated with the target that detect the acquisition radar and are designated by the scenario as acquisition jammers will jam the acquisition radar. For the acquisition radar to establish acquisition and turn on the tracker, it must detect the aircraft on three out of five successive scans, with no two consecutive failures to detect in the sequence. This requirement permits the acquisition radar to provide good initial information to the tracker. Designating detection by D and nondetection by N, the only sequences of successive scans that can establish acquisition are DDD, DDND, DNDD, and DNDND. If none of these patterns occurs in a time specified as a parameter of the radar type, the acquisition radar drops acquisition and returns the aircraft to the assignment procedure. Other reasons for dropping acquisition that have been implemented are:

1. Tracker or acquisition radar killed by an ARM.
2. Aircraft shot down by a SAM from another site.
3. Aircraft outbound and nominal interception point outside the maximum range of the SAM before acquisition occurs.
4. Tracker reassigned. There is a priority choice among aircraft, and the current assignment may be overridden.
5. Aircraft leaves the field of view of the radar, usually because of terrain masking.

If the tracker is turned on, it begins to track the target aircraft with errors determined by servo acceleration lag, target glint, and signal-to-noise ratio as modified by jamming. The tracker may be susceptible to range deception, angle deception, or both. If the jammer on the target aircraft or members of its group are designated to jam trackers, and if it is capable of producing the desired jamming technique, it employs the tactic appropriate to the tracker. Otherwise random noise is radiated.

Multipath and clutter effects are now included in RJARS. Multipath effects are calculated for search and track radars. Clutter effects are calculated for search and track radars, pulse doppler seekers, and CW seekers. Most of the multipath and clutter treatment is taken from ESAMS [Ref. 6].

The aircraft may be carrying ARMs. These are designated by the scenario as preprogrammed, in which case they may be launched only at a specified radar, or opportunistic, for which they may be launched at trackers which are operating against the indicated aircraft. Each ARM has a specified operating frequency band and a minimum and maximum range, and the target tracker must be within the frequency and range limits for launch to take place. Once an ARM is launched, it flies in a lofted trajectory at constant velocity toward its target. If the radar ceases to transmit, the ARM will continue its flight for a specified time, with the kill probability reducing linearly to zero during that time. The alternate condition in which the ARM diverts to another target is not implemented in RJARS, a limitation of the simulation. Otherwise, it will fly to the target and perform a random draw against a probability of kill to determine the outcome of the attack.

With the tracker providing information about the path of the aircraft, the SAM system will determine when a SAM should be launched. The requirement for launch is that the position of intersection of the extrapolated path of the aircraft and the straight line path of the SAM when flown at the proper heading should lie between the range of the SAM at booster cutoff and the maximum range of the SAM. All SAMs are treated as four-stage devices composed of an initial unguided booster, an unpowered coast (for most SAMs the duration of the coast is zero), a powered and guided sustainer, and an unpowered guided coasting stage. After a random draw to determine launch success, the SAM is launched, propelled by calculating thrust and drag accelerations, and guided either by a variation of lead pursuit guidance in which the SAM heading is pointed toward the projected aircraft position or by proportional navigation, each with a lag corresponding to the missile dynamics. When the SAM reaches its calculated minimum distance from the target, which calculation includes the errors in the radar guidance information, it determines a nominal miss distance by adding an error in an arbitrary direction corresponding to the error in the SAM guidance system. This point of closest approach is compared to the oriented aircraft. The aircraft is taken to be an ellipsoidal fuselage with elliptical wings.

If the closest approach point is within the aircraft, corrsponding to an actual hit, the kill probability is taken to be unity. This is a reasonable assumption for large warhead weapons, but is too favorable to the defense for small warhead weapons or artillery. If the aircraft is missed but the weapon contains a proximity fuse (shoulder-fired IR SAMs and all guns are assumed to have contact fuses, others proximity fuses), the SAM kill probability is taken to be constant for a miss distance less then a specified value (the value of the constant is the SAM warhead reliability), and to drop linearly with miss distance to zero at a value large enough that the kill probability should be negligible. Another random draw determines the outcome of this endgame. If the aircraft is killed, it is removed from the simulation, and the tracker and SAM revert to the available state. If the aircraft survives, the equivalent of "shoot-look-shoot" makes the SAM start counting down again. The process is continued until the aircraft is killed or escapes from the effective range of the tracker-SAM combination. In the latter case the assignment system takes over.

The term "SAM" may include anti-aircraft artillery. RJARS treats a gun as a special case of a SAM. The propulsion takes place only in a very short "booster" stage, chosen to give the projectile its correct initial velocity, and the trajectory thereafter is unpowered, constant mass, and unguided (drag is still simulated). The launch calculation is modified to determine the shell trajectory and find the proper aiming direction. Air density variation and gravity drop are included. The kill probability is the kill probability per burst, not per bullet. In other respects the simulation is the same.

Radar-guided SAMs may obtain their guidance information directly from the tracker via data link, corresponding to command guidance, or may use a semiactive system in which an illuminator located at the tracker shines on the target and the reflected signal is received at the seeker mounted on the missile. For a semiactive system the illuminator is turned on at launch time.

While all these complex operations are going on in the defensive system, the aircraft is performing maneuvers of its own. Under the control of the scenario, it can accelerate, climb, or turn, each at a specified rate, to a new value of velocity, altitude, or heading. Jammers can be turned on and off by the scenario, as well as by the automatically operating warning receiver. It can split from or join groups. ARMs can be launched, as can air-launched cruise missiles. The latter are treated by the simulation as additional aircraft, each with its appropriate parameters and flight maneuvers. The cruise missiles are visible to the radar only after they have been launched. Maneuver commands should not be given to cruise missiles while they are on board their respective carriers. The aircraft may also tow jamming decoys.

The aircraft will normally have a destination within the interaction field, and will then turn and proceed homeward. This change triggers the defensive system to lower the priority on that aircraft, so if it is being tracked and no SAM has yet been launched, the tracker is included in the assignment pool if an incoming aircraft is detected.

RJARS grinds through the simulation as described above until either the end of the simulation time is reached or all aircraft are shot down. Output is provided at selectable time intervals for the aircraft, radar, and missile status, positions, etc. At the end of the run, summaries are printed showing the detections of each search radar against each aircraft, tracking times of assignment, acquisition, and tracking, SAM and ARM launches and outcomes, and an overall box score.

RJARS can perform Monte Carlo operations. If a number of runs is specified, at the end of each run until the last all the scenario parameters except the random number generator will be returned to their initial values. The run will then be repeated, but the outcomes of the various random draws can be expected to be different. Statistics on the runs can be compiled.

RJARS now includes a graphics calculation. This is provided as a file in CAGIS format, and is the only nonportable part of RJARS. With the graphics, the terrain and defenses may be depicted on the screen, along with the aircraft, SAM flight paths, and their interactions. As an aircraft progresses along its path, it is initially in black. The path color changes to gray when a searcher detects the aircraft, to blue when a tracker is in acquisition, to yellow when a tracker is in track, and to orange when a SAM or shell has been launched. SAM paths are in red. An endgame miss is a small red star, an endgame kill is a large red star. Icons are used for the defenses to indicate the type of radar, SAM, or gun. Targets for the aircraft may also be shown. Iterations may be presented in sequence. The paths may be laid down all at once, or the flights may be time-stepped. The latter is usually much more descriptive. The graphics operation is an RJARS postprocessor, for which the data file is prepared during the RJARS run, and the graphics program may be run at any time.

The printed output from RJARS is extensive, and we have usually found it sufficient for our purposes. The graphics are useful for determining general characteristics of a particular simulation run.

This completes the general program description. We shall next present the analysis and show the development of the several program modules. RJARS contains 11 updating modules:

1.	UPDCK	System clock
2.	UPDAC	Aircraft positions and maneuvers
3.	UPDTR	Over-terrain visibility
4.	UPDRS	Search radars
5.	UPDSI	Sites and assignment of trackers
6.	UPDRT	Tracking radars
7.	UPDAR	Anti-radiation missiles
8.	UPDSM	Surface-to-air missiles
9.	UPDWR	Warning receivers
10.	OUTPUT	Summary outputs
11.	UPDMC	Monte Carlo

These will be discussed in detail.

N-2727-AF [Ref. 3] contains an appendix listing of the PL-I program. The C program is inherently much longer, and with the many added phenomena the listing has grown too long to be practical. Therefore, the Note has undergone an appendectomy followed by a resection (a new and much shorter appendix containing flow charts).

III. ANALYSIS SECTION

A. UPDCK--SYSTEM CLOCK

The system clock is updated at each time step. The author has generally used a time step of 2.5 seconds. This is shorter than the scan period of most search radars, but is such that the aircraft will move 1400 feet (Mach .5) to 5600 feet (Mach 2) during the step, distances large enough to be observed by the system. Since the SAMs require much better position measurement, the time step is subdivided and the aircraft position interpolated when a SAM is in flight toward the aircraft.

The clock keeps track of the simulation time and of the number of surviving aircraft. If the end of simulation time (maximum 9999 seconds) is reached, or if all the aircraft have been shot down, the clock signals the main program to terminate this simulation run, print the summaries, and continue to the next Monte Carlo sequence if necessary.

B. UPDAC--AIRCRAFT POSITION AND MANEUVERS

In the subroutine UPDAC, the aircraft position is updated and any scenario-required maneuvers are performed. If terrain effects are included (VIS = 2 or 3) the aircraft input altitude and any subsequent climb maneuvers are specified with respect to the terrain, so a constant altitude value would correspond to terrain following. The flight path may be presented in two ways, either by a sequence of maneuver commands, like the previous treatments, or by the prescription of an external flight path generator (BLUE MAX or CHAMP). The maneuver sequence technique will be described first, then the flight path generator technique.

1. Position Updating and Maneuvers

The position updating used in JARSM has been improved in RJARS. In JARSM the aircraft latitude and longitude are updated at the beginning of UPDAC, using the values of velocity and heading valid at the previous time step. If a maneuver is called or in progress the velocity, altitude, or heading is then updated, using the value of acceleration, climb rate, or turn rate, as appropriate. However, the position is not further updated. If, for example, an acceleration of 2 g (64 fps^2) is in progress, in an interval of 2.5 seconds the aircraft will travel an additional 200 feet. This may not be important to radar detection, but is far too much error for SAM guidance. Consequently, RJARS carries position updating to second order. In addition, climb maneuvers include pitching to the correct attitude.

The aircraft velocity vector in RJARS has magnitude ACVEL, pitch ACPITCH (horizontal = 0), and heading ACHDG (zero northbound, positive clockwise). Simplifying these to V, P, and H, the northward, eastward, and vertical components of velocity are:

$$V_N = V * \cos P * \cos H \qquad (1a)$$

$$V_E = V * \cos P * \sin H \qquad (1b)$$

$$V_Z = V * \sin P \qquad (1c)$$

Differentiating these gives the rectangular components of acceleration, where a dot above a variable denotes its time derivative:

$$\dot{V}_N = (\dot{V} * \cos P - V * \dot{P} * \sin P) * \cos H - V * \dot{H} * \cos P * \sin H \qquad (2a)$$

$$\dot{V}_N = (\dot{V} * \cos P - V * \dot{P} * \sin P) * \sin H + V * \dot{H} * \cos P * \cos H \qquad (2b)$$

$$\dot{V}_Z = \dot{V} * \sin P + V * \dot{P} * \cos P \qquad (2c)$$

In JARSM, and also in RJARS, the motion of the aircraft is straight and level unless a maneuver is called. A maneuver may be an acceleration, for which dV/dt is constant, a horizontal turn, for which dH/dt is constant, a climb, for which V_z is constant, or a pitch, for which the pitch rate dP/dt is constant.

The scenario specifies the maneuvers in terms of a starting time T_0, which need not be a multiple of the time step, DT, a maneuver rate (acceleration, turn rate, pitch rate, or climb rate) and a final value for velocity, heading, pitch, or altitude. The duration of the maneuver is the quotient of the difference between final and initial values and the maneuver rate, and usually includes several time steps. The time available for maneuver during each time step is designated VAR5 in the program (VAR5 is one of many dummy calculation variables and has other meanings in different parts of the program). At the time step immediately after initiation of the maneuver, VAR5 = T - T_0. For intermediate time steps, VAR5 = DT. For the final time step, VAR5 = T_0 + (maneuver duration) - T, where T is the simulation time at the last time step before the end of the maneuver.

The maximum pitch acceleration (up or down) and the maximum turn acceleration are given in g's for each aircraft type in ACLIB. If the turn rate or pitch rate specified in the maneuver leads to an angular acceleration that exceeds the maximum, it is replaced by the limiting value.

A climb maneuver is rather complex. The specified input parameters are the climb rate and the final altitude. The rate determines the pitch during the steady climb part of the maneuver. If the climb is upward, a pitch maneuver to the steady pitch at the maximum upward pitch rate (PITACUP) begins the climb, and a downward maneuver to level flight at the maximum downward pitch rate (PITACDN) completes the maneuver. Between these, the aircraft climbs at a constant rate. The time to initiate the downward maneuver is calculated before the climb begins. For descents, the pitch maneuvers are first down then up. If the difference in altitude is so small that there is no time for a steady climb at the indicated climb rate, the up and down pitch maneuvers are matched to the altitude difference.

Occasionally the maneuvers will be so small that they should be completed in less than one time step. RJARS checks such details and modifies the values of VAR5 (see above) accordingly. For example, the first pitch maneuver of a climb may take less than one time step. RJARS will complete the pitch maneuver, bring the aircraft to the proper attitude, velocity, and position, and initiate the climb maneuver with the correct value of VAR5. These techniques are applied to all maneuvers.

At the beginning of UPDAC, at every time step the horizontal components of position are updated by:

$$\Delta LAT = V * DT * \cos P * \cos H \tag{3a}$$

$$\Delta LONG = V * DT * \cos P * \sin H \tag{3b}$$

where Δ denotes "change of." During an acceleration maneuver, the velocity and horizontal components of position are updated by:

$$\Delta V = \dot{V} * VAR5 \tag{4a}$$

$$\Delta LAT = .5 * \dot{V}(VAR\ 5)^2 * \cos H * \cos P \tag{4b}$$

$$\Delta LONG = .5 * \dot{V}(VAR\ 5)^2 * \sin H * \cos P \tag{4c}$$

During a pitch maneuver, the pitch and components of position are updated by :

$$\Delta P = \dot{P} * VAR\ 5 \tag{5a}$$

$$\Delta LAT = -.5 * V * (VAR\ 5)^2 * \cos H * \sin P \tag{5b}$$

$$\Delta LONG = -.5 * V * (VAR\ 5)^2 * \sin H * \sin P \tag{5c}$$

$$\Delta ALT = .5 * V * (VAR\ 5)^2 * \cos P \tag{5d}$$

During a turn maneuver, the heading and horizontal components of position are updated by:

$$\Delta H = \dot{H} * VAR\ 5 \quad (6a)$$

$$\Delta LAT = -.5 * V * \dot{H} * (VAR\ 5)^2 * \cos P * \sin H \quad (6b)$$

$$\Delta LONG = .5 * V * \dot{H} * (VAR\ 5)^2 * \cos P * \cos H \quad (6c)$$

During a climb maneuver, the altitude is updated by:

$$\Delta ALT = V_Z * VAR\ 5 \quad (7)$$

These formulas hold for the increments in velocity vector and position vector during the maneuver. An additional correction is required during the final time step to take account of the fact that after the maneuver is completed, the velocity vector continues at its commanded value rather than its value at the beginning of the time step. This correction adds to the latitude change during the final step of acceleration

$$\Delta LAT = V * VAR\ 5 * (DT - VAR\ 5) * \cos H * \cos P \quad (8)$$

and corresponding terms for the other position terms.

A note on units is in order. Both JARSM and RJARS employ a mixed system of English units. Latitude and longitude are in nautical miles, altitude in feet. Horizontal and total velocity are in knots, vertical velocity in feet per second. All angles are in degrees and all times in seconds. As a result, factors of 6080 (feet per nautical mile), 3600 (seconds per hour), and DR=π/180 (radians per degree) are scattered all through RJARS, and anyone who wishes to make modifications must be wary.

2. Flight Path Generators

RJARS is prepared to accept input flight data from either the fixed-wing flight path generator BLUE MAX or the helicopter flight path generator CHAMP. They work in basically the same manner. The CAGIS mapping program is used to set up the terrain over which the path is to be flown. The user selects a starting point, then a next point along the path. In the terrain following mode, the path generator will attempt to fly from one point to the next at constant altitude above the terrain, going directly over obstacles. In the terrain avoidance mode he will look ahead and either go over or around the obstacle. The aircraft dynamics are used to find the actual path under the conditions indicated. The path is continued through an appropriate number of points to the destination and return. The use of these path generators is strongly dependent on the ability of the simulator pilot, who must understand both the dynamics of the aircraft and the tactics that may be required for the mission, lest he crash or overstress on route. For example, flying a helicopter five feet off the ground through trees at a speed of 150 knots is permitted by the flight path generator, but is unlikely in real life. These flight path generators, whose operation has many arcane qualities, have proved extremely valuable.

The output of the flight path generator is a file that includes for each path an identification number, a time step, and the aircraft position vector (latitude, longitude, and altitude), its velocity vector (velocity, pitch, and heading), and its attitude vector (nose pitch, bank, and yaw). For helicopters, it also contains a hiding parameter (0 if fully exposed, 1 if only rotor exposed, 2 if hiding). For the special case of tilting rotor vehicles, the tilt angle is presented.

There may be several flight paths used in the simulation. RJARS employs the convention that helicopter flight paths, obtained from CHAMP, have an identification number greater than 100. Helicopter types apply the same convention. The flight paths may have different durations and different initial delays. The primary purpose of the initial delay is to permit several aircraft to be flown down the same path without having to construct new visibility tables. If the simulation time exceeds the duration of the path, RJARS uses the PITCH maneuver to bring the aircraft to level flight, then continues the path straight and level over the terrain with the velocity and heading of the last point on the path. Similarly, for times before the initiation of the delayed path, RJARS extrapolates the path backwards from the initial point straight and level. If a path goes off the terrain, the aircraft is removed from the simulation with the announcement that it has been TERMINATED WITH EXTREME PREJUDICE. This is to avoid trying to calculate height above nonexistent terrain, which will go outside the limits of the terrain array.

3. Nonpositional Maneuvers

In the flight maneuvers section of the scenario, each maneuver for each aircraft is written as a single line containing four values--the time of the maneuver, a character expression (five or fewer letters) identifying the maneuver type, and two parameters (X and Y) that give the maneuver specifics. The maneuvers for a specified aircraft are sequenced by the maneuver time (several maneuvers may begin at the same time), and the last maneuver for each aircraft must have the time 9999. The sets of maneuvers for each aircraft are then sequenced by aircraft number (not specifically included in SCENA), so first all the maneuvers for aircraft 1 appear, then all those for aircraft 2, up to the last aircraft. If an aircraft has no maneuvers (an unlikely event), the 9999 line must still appear. In addition the flight maneuvers section includes a dummy aircraft, NAC + 1, where NAC is the number of aircraft. This dummy, which must be present, represents connection and cutting of the communications systems, and was placed here because it has the same input pattern as flight maneuvers. It is discussed in the command section UPDSI.

Besides the positional maneuvers ACCEL, CLIMB, PITCH, and TURN, the aircraft has several mission-related maneuvers. Those are:

1. ON   Turns the aircraft jammer on. Used only

	if at some earlier time the jammer has been turned off.
2. OFF	Turns the aircraft jammer off. Used if the scenario designer wishes to maintain radio silence during a point of the flight, or to simulate jammer failure.
3. HOME	Flag to indicate the aircraft has reached its destination and is returning to base. Acts as trigger on the defensive system to reduce priority.
4. SPLIT	The aircraft splits from its group (parameter X) and will jam only those radars involved with itself.
5. JOIN	The aircraft joins a group (parameter X) and will jam all radars involved with the group. SPLIT and JOIN simulate formation flying and flight command operations.
6. VIS	Aircraft becomes visible to a radar (X), and will be until a time (Y). Used to simulate terrain effects when operating in the "terrain not included" mode (VIS = 0).
7. BLANK	Used for air-launched cruise missiles (ALCM). ALCM is on board aircraft (X) and is invisible to all radars. After a missile has been placed aboard a carrier aircraft, it will maintain the position of its carrier. Maneuvers should not be given to a cruise

	missile while it is blanked.
8. SEP	Used for ALCMs. Aircraft (X) has launched the ALCM, which can now be observed by radars.
9. DECOY	Release a towed decoy jammer to a backward displacement X (ft). Jammer on decoy is of type Y.
10. ATTK	Attack command. Aircraft (or ALCM) attacks its target at latitude (X) and longitude (Y). If information about kill probability of weapons is available, a kill maneuver command can be added.
11. BLUMX	Fixed-wing aircraft is to follow BLUE MAX flight path X with time delay Y. This is given with TIME = 0. The flight path, previously read in, is associated with the aircraft, which will follow the stored flight path values for position, velocity vector, and attitude.
12. CHAMP	Helicopter follows CHAMP flight path X with time delay Y. See BLUMX. The hiding state of the helicopter is included in the data.
13. HIDE	Helicopter goes into hiding at altitude X above ground.
14. SENSE	Helicopter exposes its sensor (if any) at altitude X above ground.
15. ROTON	Helicopter exposes its rotor only.
16. POPUP	Helicopter climbs vertically to altitude above

	ground X with vertical velocity Y (fps).
17. BLINK	Establish aircraft X as partner, using blink jamming technique Y.
18. BLKPR	Set blink jamming period Y with partner X.

In the actual C program, the names of the maneuvers are supplemented with sufficient x characters to form a five-character word.

RJARS is capable of handling five types of jamming decoys. The decoy is itself an aircraft of type 13. The scenario specifies that there are NDECOY decoys on each carrier aircraft, where NDECOY may be zero. Only one decoy per aircraft is towed at a given time. At the beginning of the simulation, a decoy is deployed via the maneuver DECOY (see above). If the decoy is killed by a SAM that has been fooled into attacking the decoy instead of the carrier aircraft, another decoy will be deployed, until there are none left. The jamming techniques available to the decoys will be discussed under tracking radars.

The position of the aircraft is printed out at intervals determined by a multiplier IPA. If DT = 2.5, IPA = 4, the position will be printed every 10 seconds. If a position maneuver is in progress, the maneuver time and current value of velocity, heading, or altitude is printed at each time step during the maneuver. Mission maneuvers are printed at their time of occurrence.

C. UPDTR--OVER-TERRAIN VISIBILITY

The manner in which terrain is treated by RJARS is controlled by the variable VIS. It may take on five values, with the following meanings:

1. VIS = 0	No terrain, smooth earth.
2. VIS = 1	Visibility during simulation read from ACVIS file.

3. VIS = 2	Preparation of visibility file, ACSGT, from aircraft flight paths, radar positions, and TERRA file.
4. VIS = 3	Visibility over terrain calculated during simulation using TERRA file.
5. VIS = 4	Calculate the ridges for the radar laydown.

If VIS = 0, the visibility calculations and over-the-horizon determination are performed using a smooth spherical earth with radius equal to 4/3 of the earth radius, a factor that takes into account the average effects of refraction on the radar line of sight. For the other cases, the terrain is used to control the visibility between aircraft-radar pairs.

It is simplest to first explain the calculations when VIS = 3. The input file TERRA contains the locations and heights of the many terrain elements contained in the field of action of the simulation. The locations may be in RJARS rectangular coordinates, or may be global values in the form of degrees and fractions of degrees (not degrees, minutes, and seconds). The locations and heights may be derived from maps, from preexisting terrain files, or by using CAGIS. As presently designed, two terrain elements are stored on each line of TERRA. If the field of action is 50 miles by 50 miles and the resolution element is 1/4 mile, there will be 40,000 elements, occupying 20,000 lines, so evidently TERRA can be very large. The terrain field and the aircraft flight paths must be so related that no aircraft ever comes into a border element of the terrain.

As the simulation proceeds, the aircraft are flown along their prescribed paths. At each time step, the azimuth and elevation of the line of sight from each radar to each aircraft is calculated. This line of sight will pass over several terrain elements as it extends to the range of the aircraft. Beginning at the radar, the distance along the line of sight is incremented in steps equal to half the length of the diagonal of a terrain element. All terrain elements are the same size (miles or degrees). At each point, the height of the line of sight above sea level on the spherical earth is calculated by first finding the height above the plane surface that is tangent to the radius vector from the center of the earth to the radar, then lowering by the square of the ground range divided by twice the modified earth radius. The height-finding subroutine ZTERN is called to find the height of the terrain at the point beneath the calculation point. If the terrain height exceeds the test point height by ten feet (an arbitrary margin included to provide some effect of aircraft size), the line of sight is masked.

It is assumed in the calculations that the field of action is sufficiently small compared to the modified earth radius (4586 nm) that only first-order terms in the ratio need be kept. This should be valid at least for dimensions of 200 nm.

Each line of sight is tested along its length to determine masking. If a line of sight is masked, that aircraft-radar pair is excluded from the simulation calculations. Memory is provided for acquisition or tracking radars, which are not treated as masked until three scans or settling times have elapsed.

The modes VIS = 1 and VIS = 2 are complementary. If VIS = 1, the file ACVIS is read to provide a list of times for each aircraft-radar pair during which they are mutually visible. The mode VIS = 2 creates the file ACSGT, equivalent to ACVIS, and thus acts as a preprocessor for VIS = 1.

With VIS = 2, the aircraft are flown along their paths, but no radars or missiles are activated. At each time step, the masking calculations are performed as described. Initially, all pairs are assumed masked. A visibility matrix with three arguments (the aircraft-radar pair and counter VISKEY for that pair), is initialized so all on-times are 9999 and all off-times are zero. If a line of sight becomes unmasked, the counter VISKEY for that pair is incremented, the on-time is stored, and the off-time is set to 9999. If at a later time the line becomes masked again, the off-time is stored. This procedure is continued to the end of the simulation. Then the visibility matrix is read into the file ACSGT in order of increasing on-time (actually, nondecreasing on-time, since several pairs may have the same on-time). A terminator line with an on-time of 9999 is automatically added to the file, since the input file ACVIS (see below) always reads one line further than the last line whose on-time equals the current time.

If VIS = 2 has been run, the terrain may be included in the simulation by running VIS = 1. The file ACSGT should be copied to the input file ACVIS. Then when RJARS is run, the visibility data for each pair will be read in at the time that visibility begins. The simulation proceeds with the aircraft-radar horizon limits determined by the visibility data from the file.

If ACVIS is prepared as shown, all on times will be multiples of the time step. It is also possible to prepare ACVIS directly from maps or CAGIS if the terrain field is small or smooth. In this situation the on times may be arbitrary.

At RAND, ACVIS is normally prepared by CAGIS, so the mode VIS = 2 is not required. However, users of RJARS who do not have CAGIS available, or who are not concerned with a specific terrain region, can use VIS = 2 to calculate visibility without the intercession of CAGIS.

If only a single run is desired, or if the terrain field is small, it is reasonable to use VIS = 3. In most cases, if many runs are desired, or different jammers or radars are to be considered, the preprocessor VIS = 2 should be employed, so that the later runs with VIS = 1 do not have to calculate the terrain masking effects.

The mode VIS = 4 is used to calculate ridges for clutter or optical systems. This is normally performed in CAGIS, but RJARS also has the requisite capability. For each ground equipment, calculations are performed at a set of angles determined by the parameter nrdgangle (number of ridge angles, defaulted to 37, corresponding to an angular spacing of 10 degrees). Along each direction, distance is stepped in small units, and the visibility to the radar of the ground at each point is ascertained. Because of the variable terrain height, the ground may go in and out of visibility. Each time it goes out of visibility, the range and depression angle are stored. When it reappears, at the previous depression angle, the range is stored. The process is continued out to the terrain boundary. The number of ridges in any direction is of course unknown at the beginning of the simulation.

Multipath and clutter effect calculations require a knowledge of whether the ground is visible to the radar or seeker. Optical equipments use ridge data to determine if the target is seen against a sky or ground background. The ridge data file (RIDGE) is required if any of the flags ICLUT (clutter included), IMULT (multipath included), or IOPT (optical equipment in use) is set. The read-in scans the RIDGE file twice, the first time to determine the number of ridges in each direction, the second time to dynamically allocate space and read the ridge data.

If CAGIS is available, it provides the ridge data file. If not, for a given defensive laydown first run RJARS with VIS = 4. The ridge data are stored in the file RDRDG, which should be transferred to a permanent location. When the ridge data are required for the simulation, they should be copied to the file RIDGE, which is the data input file for use of ridge data. Thus, VIS = 4 is a preprocessor for RJARS, and need be run only once for each defensive laydown. Again, like VIS = 2, it should be run before using VIS = 1 for the actual simulation.

D. UPDRS--UPDATE SEARCH RADARS

In RJARS radars are divided into six categories. Each radar is identified by a four-character code. The first character may be L (long-range search), H (height finding), A (acquisition), T (track), Q (IR), or G (gun). The letter Q is used rather than I because I is used for the illuminator state for semiactive missile launchers. The height-finders' peculiar scanning and nodding motion has not been implemented in RJARS, so an option HF has been included in the scenario. If HF = N, then the height-finders are excluded from calculation. Also, no height-finders need be included in the scenario. If height-finder-type radars are included, they are treated just like long-range search radars. The radar theory used in RJARS is based on Refs. 4 and 5.

The data pertaining to each radar type are contained in file RDLIB. See the user's section for instructions on preparing RDLIB . The scenario contains the location, type, and site identification of each radar. The radars may be in any order in the scenario. There are many variables that are associated only with search radars, and many others, including all SAM variables, that are associated only with trackers. The PL-I version of RJARS carries the radars in two lists, one for trackers and one for all others, thus reducing the required working space considerably. Because of the multifunction radar concept and the dynamic allocation, this double listing has not been incorporated into the C version. It probably will be in the future.

After the radar data have been entered, certain calculations are performed to set radar variables that are retained throughout the simulation. The search radars are initially turned on and are oriented in random directions. The frequency of each radar, which is specified by RDLIB as being between an upper and lower bound, is set to a random value within that frequency range. The search radars then begin scanning clockwise at their specified scan rates, and will continue to scan unless they are killed by an ARM.

Optical systems search a sector rather than spinning. The subroutine SECTSCAN moves the direction of observation in a sawtooth pattern across the sector of responsibility. Initially viewers are at the lower scan limit. The subroutine INBEAM determines if the sawtooth scan has passed over the target during the time step.

1. Signal and Jamming Power Analysis

The signal power received from the aircraft by the radar at the time when the main beam of the radar passes over the aircraft is:

$$S = \frac{(ERP)_R*G_R*C^{2*}\ \sigma_T}{(4\pi)^3*F_R^2*R_{RT}^4} \qquad (9)$$

The variables in this equation are:

$(ERP)_R$	Effective radiated power of the radar
G_R	Radar antenna gain on axis
L_R	Losses in the radar receiving system
C	Velocity of light
σ_T	Scattering cross-section of the target
F_R	Radar frequency
R_{RT}	Range from the radar to the target

This may be rewritten in the form:

$$S = (RPROD)_R\ \sigma_T/R_{RT}^4 \qquad (10)$$

where σ_T is in square meters, R_{RT} in nautical miles, and the multiplier RPROD, held through the simulation, is:

$$RPROD = 2.32*\ 10^{-8}*ERP(W)*G*L/(F(MHZ))^2 \qquad (11)$$

where the subscript R is implied.

Equation (9) represents free space propagation, with no additional attenuation of the signal. Rain attenuation, which can be significant at the higher radar frequencies, is not modeled in RJARS, so the simulation corresponds to frequencies below 10 GHz, or clear weather conditions for frequencies above 10 GHz.

In the absence of jamming, the noise power in the receiver is:

$$(MJS)_R = k*T*(NF)_R*(BW)_R \tag{12a}$$

$$= 4*10^{-14}*BW(MHZ) \tag{12b}$$

where k is Boltzmann's constant, T the receiver temperature, NF the receiver noise figure, and BW the receiver band width. In going from (12a) to (12b) the effective receiver temperature T*NF has been set to 3000 degrees Kelvin. The actual temperature varies from receiver to receiver. However, for any well-designed radar the signal is large compared to receiver noise when the target is within the operating range of the system. The receiver noise provides a floor for the undesired signal amplitude when the radar is jammed, but any effective jammer will produce at the radar a power large compared to receiver noise. Thus the receiver noise temperature, which in any case does not vary over a wide range (600 degrees Kelvin is very good, 10000 degrees Kelvin is very poor), is not a critical parameter.

The signal will be processed by the receiver, which will integrate over the number of pulses received from the target. The receiver threshold will be set to a level determined by the false alarm rate. In JARSM, this threshold setting was an input parameter, which had to be set by the scenario writer. In RJARS, we have used the theory of detection for search radars (Refs. 4, 5) and fitted the curve for probability of detection 0.5, false alarm rate 10(-6)(Ref. 4, page 2.22, Fig. 9) by the expression:

$$MDS(dB) = 11.2 - 8.45*\log_{10}N + 0.75*(\log_{10}N)^2 \tag{13}$$

where N, the number of pulses integrated

$$N = (PRF)_R*(SCN)_R*(HB)_R/360 \tag{14}$$

If the false alarm rate were raised to 10(-4), the threshold MDS would be lowered by 1.5 dB (the constant 11.2 in equation 13 would become 9.7). This is not a significant variation.

In Eq. (14), PRF is the radar pulse repetition frequency in pulses per second, SCN is the radar scan period in seconds, and HB is the horizontal beamwidth between half-power points of the radar antenna. The receiver is set at a threshold MDS dB above the noise floor MJS.

When jamming is present, the power received from each jammer is:

$$P = \frac{(ERP)_J * G_R * L_R * C^2 * L_P * (BW)_R * (EFF)_{JR} * (SDL)_{JR}}{(4\pi)^2 * F_R^2 * R_{RJ}^2 * (BW)_J} \qquad (15)$$

where the added variables are:

$(ERP)_J$	Effective radiated power of the jammer
G_{JR}	Gain of the jammer antenna in the direction of the radar
L_P	Loss due to polarization difference between radar and jammer--usually 3 dB, corresponding to linear polarization of the radar and circular polarization of the jammer
$(SDL)_{JR}$	Sidelobe attenuation of the radar in the direction of the jammer, relative to main beam gain
R_{RJ}	Distance from radar to jammer
$(BW)_J$	Bandwidth over which the jammer is radiating noise
$(EFF)_{JR}$	If the jammer is employing deceptive tactics,

$(EFF)_{JR} = (BW)_J/(BW)_R$, so all the
jammer power is in the radar receiver band.
Otherwise $(EFF)_{JR} = 1$

Many airborne jammers use multiple antennas to achieve coverage without scanning. Each antenna provides coverage in a particular direction. The combined pattern is approximated by a "rosette" antenna pattern, which contains JMNLEAF lobes. All lobes have the beamwidths JMNHB and JMNVB in the horizontal and "vertical" planes and are offset by an angle JMDIP in the vertical plane. The first lobe is offset from the forward direction by an angle JMBS, and the succeeding lobes are centered on directions displaced from the first by multiples of an angle JMPHIN = 360/JMNLEAF. The antenna gain is calculated from this rosette pattern.

Among the radar's input parameters is the jamming tactic to which it is susceptible, as discussed in the general simulation description. Since all deception techniques involve repeating the transmitted signal, thereby matching the band width, the indicated effectiveness factor is included in the jamming power. RJARS has additional capabilities for determining jamming effectiveness, which will be discussed under tracking radars.

To calculate the sidelobe level, the antenna pattern must be known. In RJARS the basic antenna pattern is an elliptical pencil beam. The sidelobe level SDL is calculated by the subroutine ANTPAT1, which has as arguments the azimuth angle off axis (PHI), the elevation angle off axis (THETA), the horizontal and vertical beamwidths between half-power points (HB and VB), and the backlobe level BL. The elliptical scaled off-axis angle X is given by

$$x = \left[\left[\frac{\sin(\mathrm{THETA})}{\mathrm{VB}} \right]^2 + \left[\frac{\cos(\mathrm{THETA}) * \sin(\mathrm{PHI})}{\mathrm{HB}} \right]^2 \right]^{1/2} \qquad (16)$$

In terms of X, the pattern is gaussian for X less than 1.2, constant from 1.2 to 2 (filling in the first null), then follows the envelope of the many-lobed pattern out to the backlobe level. In detail;

$$SDL = \min(12.041 * X^2, 17.526) \quad X < 2 \tag{17a}$$

$$= \min(10 + 25 * \log_{10} X, BL) \quad X > 2 \tag{17b}$$

Equation (17b) is the standard pattern used by the CCIR (International Radio Consultative Committee) to characterize the sidelobe pattern of communications satellite antennas, which are similar to radar antennas. The parameters HB, VB, and BL are inputs from RDLIB. The far sidelobe level BL is often about 30-35 dB, but careful sidelobe design may make it lower.

This pattern corresponds to a single pencil beam. However, many search radars use multiple beams, stacking them in elevation to achieve sharper vertical resolution while maintaining broad coverage. Hence, search radar patterns are calculated by a pattern function ANTPATS, which combines the patterns of NSTACK individual beams, each having the vertical beamwidth NVB, with the first elevated upward by a half beamwidth, and the successive lobes having centers separated by one beamwidth. Above the highest lobe, the pattern is completed by a cosecant square function. The value of NSTACK is stored in RDLIB.

2. Jamming Sequence

The first operation in UPDRS is to determine if jamming is present, and if it is to set the backlobe canceller and present the strobe widths. The backlobe canceller is a practical device, a low-gain antenna slaved to the high-gain radar antenna. Jammers that are at angles far from the antenna boresight angle will be enhanced in the receiver associated with the low-gain antenna, relative to the high-gain antenna, and thereby identified. The gain of the main receiver is set so these signals do not appear on the display.

Each aircraft keeps a running identification of which radars it is jamming in each of the frequency bands of the jammer. This information is used to determine the power received at the radar. The antenna direction is updated at each time step by adding to it a value RSANTCH, the angle through which the radar turns in a time step. A subroutine LINSGHT, which is used in many places in RJARS, calculates the range, elevation, and azimuth from the radar to the target, and if terrain is not included in the simulation (VIS = 0), determines if the aircraft is beyond the horizon to the radar.

The aircraft sequence is run through twice. The first time, the power from those jammers that are in the backlobe of the radar is summed and the resulting total used to get the receiver threshold. The second time through, the factor by which the jammer power exceeds the receiver threshold is determined, and the antenna pattern formula (Eq. (17)) is inverted to find the angle at which the jamming signal level blends into background. This is the strobe width, which is then printed out. A scenario-set parameter IP controls the printout rate (each IP'th time step will be printed).

3. Detection Probability

Target detection probability is the next quantity calculated. The set of aircraft is run through again. It is determined at each step whether the aircraft is within the "effective" field of view of the radar. Here "effective" means that the aircraft is within the physical horizon of the radar (the flag RDACOH marks this), and whether it is within the maximum range at which the aircraft could be detected by the radar. This latter range is the product of the radar's maximum range against a 1-m^2 target, an input parameter, and the fourth root of the nose-on cross-section of the aircraft in square meters. For the special case of low-observable aircraft, marked by the setting of the flag ILOBS in the scenario, the cross section in the horizontal plane, averaged over azimuth, is used instead of the nose-on cross section. This modification was installed when it was found that RJARS was waiting too long for low-observable aircraft with very low nose-on cross section, even when they were being observed from another direction. If the aircraft is within the effective field of view, it is tested to see if the main beam of the radar passed over the aircraft during the last time step. If it did, the signal power and jamming power are calculated, and the probability of detection is found.

RJARS radars are all equipped with moving target indicators (MTI). The radial component of target velocity is calculated, yielding the doppler frequency. A two-stage delay line simulator provides attenuation proportional to the fourth power of the doppler frequency when it is below the doppler filter threshold determined by the pulse repetition rate of the radar. The clutter signal reduction for search radars is limited by scanning noise, which is proportional to the square of the number of pulses on the target. Pulse doppler systems with semiactive seekers may use a very high PRF. For such systems, eight times the width of the doppler filter is used for the critical frequency, instead of the PRF. This appears to closely match practice.

3a. Radar Cross Section

The radar cross-section treatment in RJARS is more extensive than in JARSM. The variable NDOF in the scenario determines whether the aircraft attitude is calculated. If NDOF = 3, no attitude information, the radar cross section is stored as a function of aspect angle only. If NDOF = 5, attitude information, the cross section is stored as a function of both defining angles of the line of sight from the radar to the oriented aircraft. Further, RJARS permits variable resolution for the cross section data. Two numbers, ACRCSAZRES and ACRCSELRES, which may differ, are stored in the data file ACRCS for each aircraft, and the data are stored at that angular resolution. The units of radar cross section are dBsm, decibels with respect to one square meter.

The radar cross section of a helicopter is calculated in two ways, associated with its motion over the ground and with the spinning rotor. As mentioned before, the helicopter consists of a body, a rotor, and perhaps a mast-mounted sensor. As far as the flight of the helicopter over the ground is concerned, these may be added to form the total cross section, which total is used to calculate the signal to which the doppler filtering is to be applied. In addition, the rapidly spinning rotor produces a doppler high-frequency signal, which is above the doppler frequency threshold and is not reduced by filtering, even when the helicopter is hovering. This high frequency signal is typically 8 dB below the main low-frequency rotor signal.[1]

RJARS provides helicopter visibility modes via the parameter IHIDE, which is 0 for fully exposed, 1 for only the rotor showing, and 2 if only the mast-mounted sight is showing or if the helicopter is fully concealed. The equivalence of the latter configurations arises from the observation that if the helicopter is in a condition with only its sight showing, it almost certainly is hovering. The sight then looks like a small stationary sphere, and should be indistinguishable from clutter to either a radar or an optical system.

Two formulations of detection probability are used in RJARS. The first, characterized by the scenario parameter PRBTYP = 0, is identical to that in JARSM. The signal-to-noise ratio is compared to the receiver threshold. If it is less than -9 dB the detection probability PD is 0. If it is greater than +9 dB, the detection probability is 1. For intermediate values, PD is fitted with the form:

$$PD = (1 + \sin(((S/N)dB - MDS\ (dB))*\pi/18))/2 \qquad (18)$$

The second, or "cookie-cutter" detection probability type, for which PRBTYP = 1, tests whether noise or deception is being used for jamming. If noise is used, and the signal-to-noise ratio exceeds -6 dB, then PD = 1. If deception is used, and the signal-to-noise ratio exceeds -3 dB, then PD = 1. Otherwise PD = 0. This treatment permits calculation of clearly defined "burnthrough ranges."

---

[1] The author is indebted to RAND colleague Howland Bailey for this information.

The variable PRBTYP is also used to select the table for evaluation of jamming effectiveness as determined by field data. This will be discussed under tracking radars.

The detection probability is compared to a random number, and the resulting detection choice established. As discussed in the general description, and also in Ref. 1, detection is established if two of three consecutive scans yield successful results. Once the target has been detected, the fact of detection is held until the time TDROP, which is the sum of the present time and the smaller of RSTMAX (a library input chosen as a likely value for the maximum time a radar will retain memory of a target) or the product of the radar scan period and the number of scans with successful detection. When and if T exceeds TDROP, corresponding to loss of detection, the time and range are stored for the summary and the information printed.

3b. Measurement Errors

If the detection is successful, the next step is to calculate the measurement errors. Sources of error included in UPDRS are glint and noise. The theory of glint is given in Ref. 4, Chapter 28. The time of integration of a search radar is usually long compared to the rate at which the returned signal fluctuates, and the theory thereby may be considerably simplified. The effective dimension of the aircraft (ACGLN) is taken as the larger of the projected wing and body dimensions, which in turn are approximated by multiplying the square root of the broadside radar cross section by the sine or .65 times the cosine of the aspect angle. This approximation comes from empirical fits to published data on aircraft dimensions. The center of radiation of the signal fluctuates by about one quarter of the effective length, and the corresponding azimuth and elevation fluctuations are:

$$\Delta AZ = .02166(GLN(m))/R\ (n\ mi) \qquad (19a)$$

$$\Delta EL = \Delta AZ/4 \qquad (19b)$$

The error due to integrated signal-to-noise ratio is found from Ref. 5, pages 46 and 51, as:

$$\Delta R = C*PW/4(S/N*NPLS)^{1/2} \tag{20a}$$

$$\Delta AZ = (.49 + .58\sqrt{S/N})*HB/(S/N)*(NPLS)^{1/2} \tag{20b}$$

$$\Delta EL = (.49 + .58\sqrt{S/N})*VB/(S/N)*(NPLS)^{1/2} \tag{20c}$$

where the limiting values for small and large S/N for angular accuracy given in Ref. 5, p. 51 have been combined. Here PW is the radar pulse width, NPLS is the number of pulses integrated, the same as N of Eq. (14), and the other symbols are as before.

The values of Eqs. (19) and (20) are the root mean square variances of normal distributions. A subroutine NORMV produces two normally distributed independent uncorrelated variables per call. It is used thrice to combine the errors from Eqs. (19) and (20) to obtain the errors as sampled at each time step. The resulting range, azimuth, and elevation errors are then combined to form the total position error.

4. Output and Detection Table

The signal, signal/noise, nominal position, and position errors are printed out under the control of the scenario parameters JREP, PRNTMOD, and IP. JREP is the number of the Monte Carlo repetition currently being run, and the detailed radar data are printed only during the first run. PRNTMOD determines the output format. If PRNTMOD = S, then a heading is printed, followed by all the data per radar/aircraft pair on a single line. Otherwise, the radar data have a descriptive word with each value. As before, IP sets the time step at which radar data are printed.

RJARS maintains a detection table, similar to but more extensive than JARSM. Each time an aircraft comes into the field of view of a radar, the occasion is marked by incrementing the parameter ISF, and the time and range are stored. Each time during that pass that the radar establishes detection of the aircraft, the parameter DTISD, which has as arguments radar, aircraft, and ISF, is incremented, and the time and range also stored. These variable arrays are dimensioned so the aircraft may make srchno (initialized to 10) separate appearances in the field of view, and be detected three times on each appearance. So many appearances would correspond to rapidly fluctuating terrain and low-flying aircraft. A normal scenario could expect to have mostly single detections per pass, but jamming might cause loss and recovery of signal if the aircraft follows a complicated path. Other quantities in the detection table are the flags DTFLG (detection this pass), DTNEW (new detection this pass), DTPSS (established detection this pass), and DTPSSL (established detection last pass), which last flag is used to establish loss of detection now; and the hand-off variables DTRNG, DTBRG, and DTEL. The values, including errors, are sent by the radar to its site.

UPDRS also has routines that apply to acquisition radars, either dropping acquisition or turning the tracker on. The conditions for activating these routines have been presented in the general description.

5. Multipath and Clutter

The multipath and clutter calculations in RJARS are closely based on those in ESAMS (Ref. 6). For search and track radars, clutter reduces the effective signal-to-noise ratio and produces errors in the calculated position of the target. It also puts errors into pulse doppler or CW seekers. Multipath may increase or decrease the received signal strength, and will produce elevation errors. These effects will be described, mostly on a qualitative basis.

For search or track radars, clutter comes mostly from the region of the ground under the target. The lateral extent of the clutter patch (along the ground perpendicular to the line of site) is approximately the product of the range and the antenna horizontal beamwidth. The longitudinal extent (along the ground parallel to the line of sight) is approximately the product of the pulse width and the velocity of light. The latter product is usually small compared to the rate of change of the ground properties, so the clutter region can be viewed as a thin strip on the ground perpendicular to the line of sight. This description is limited to the condition where the longitudinal extent given above is small compared to the product of the range and the vertical beamwidth, a condition usually satisfied except for very close targets. The subroutine FACE examines the clutter patch and finds which terrain elements are included in it. For them, the terrain type is found and the reflection coefficient is calculated. The data necessary to calculate the cross section per unit area as a function of frequency and angle of incidence are stored in the file MCLUT. The subroutines CLUTTER, POWER, and RESPNS find the actual clutter power returned from each element.

For clutter power to actually reach the receiver, the clutter patch must be visible. The visibility may be modified by refraction or diffraction effects. At present these are not modelled in RJARS. The file RIDGE, previously described, contains the data for these visibility calculations. The range to the patch is compared to the ridge data, interpolated to the proper angle. Since the number of ridges along the bounding lines of the target sector may be different for the two bounds, the interpolation process is complicated. If the ground is visible, then its slope is examined to make sure that it is not in a self-shadowing configuration (tilted away more steeply than the angle of incidence).

If the ground is visible, the clutter power is calculated using the indicated subroutines. The clutter cross-section per unit area of ground is determined at each relevant terrain element. This cross section is a function of the frequency FRQ (MHz), the terrain type, and the angle of incidence (THETA). The equation giving the cross-section SIGMA, derived from ESAMS, is:

$$\mathrm{SIGMA} = \mathrm{A}*(\mathrm{FRQ}/1.5\mathrm{E}4)^{.8}*(\mathrm{THETA} + \mathrm{C})^{\mathrm{B}}* \\ \exp(-\mathrm{D}/(1 + \mathrm{ROUGH}*\mathrm{FRQ}/3\mathrm{E}5)) \qquad (21)$$

The parameters A, B, C, D and ROUGH, the roughness in cm, are stored in the file MCLUT as functions of the terrain type.

The cross-section per unit area for each element is multiplied by the pulse width and the velocity of light and the length of the element in the direction perpendicular to the line of sight. It is then multiplied by the square of the antenna gain in the direction of the element, corresponding to two-way propagation, to yield the clutter power at the receiver. The power is summed over the active terrain elements. This power is filtered using the doppler filters described above (actually the signal is reduced relative to clutter if the doppler frequency of the signal falls into the filter notch). The filtered clutter power is added to the noise plus jamming to give the effective signal-to-noise ratio of the system, and the mean azimuth and elevation of the clutter, appropriately weighted, are combined with the signal to give errors in the indicated position.

For low-flying aircraft, it is quite possible that the clutter signal will be strong enough to mask the signal from the aircraft. This is only likely for pulse (not pulse doppler) radars and, because of the filtering, aircraft flying in a direction perpendicular to the line of sight. The effect will show in RJARS in the calculation of the detection probability, for which the resultant signal may drop below the effective minimum detectible signal. Sometimes, if the aircraft is distant but visible, the ground at the same location will not be visible, permitting detection, and then as the aircraft approaches the ground comes into visibility and the aircraft drops out of detection.

Multipath effects are also simulated in RJARS. Multipath is a phenomenon involving reflections of the returned signal from the target against the ground. If the angle of incidence of the ray from the target to the ground point and the ray from the radar to the ground point make equal angles with the local slope of the ground, a condition known as specular reflection, the ground signal may be very strong. The representation of multipath in RJARS is also derived from ESAMS.

Multipath signals come from a region on the ground surrounding the specular point or points. If the terrain is flat, this region, known as the first Fresnel zone, is a long narrow ellipse, the size and shape of which are determined by the condition that the total path length of the rays from radar to ground and target to ground shall not exceed the total path length of the rays to the specular point by more than a half wavelength. The program steps through the points on the ground below the sight line to the target. For those points that are visible and not self-shadowed to either ray, the angles of the rays with the local slope of the ground in the direction paralleling the sight line are found and tested for equality, thereby determining the presence of specular points. The size of the Fresnel zone is found from the elliptical formulas treating the return as coming from the tangent plane at the specular point. If the ground is rapidly varying, this may be a bad approximation. However, the complication of finding the actual shape of the region on the ground using the ray length criterion is so great that the simpler procedure has been retained.

The reflection coefficient at the specular point is found using the Fresnel reflection coefficients, which are calculated from the dielectric properties of the ground. The real and imaginary parts of the dielectric constant are stored as functions of terrain type in MCLUT. They are actually functions of frequency, but the variation over the microwave region is small enough, and the values uncertain enough, that constants have been used. The reflection coefficient depends on the radar polarization, stored in RDLIB, and on the angle of incidence. For most configurations of interest the angle of incidence is very small and the reflection coefficient is close to -1.

The total multipath signal is found by multiplying the reflection coefficient by the area of the Fresnel zone and by the square of the antenna gain in the direction of the specular point. Again, if the zone is long, the antenna gain should have been weighted over the zone. This is too complicated to consider.

The multipath signal is combined with the direct signal from the target using the proper phase, determined by the difference between the lengths of the direct ray from the target and the reflected ray from the specular point. As the aircraft travels, this phase difference may change rapidly. Fluctuations caused by multipath are well known in radar phenomenology. The signals may add in phase, causing an increase in detection range, and then as the aircraft flies the return goes in and out of visibility as the interference of the rays changes from constructive to destructive.

This combined multipath signal is used to determine the target detectibility, and the effective azimuth and elevation of the weighted combined signals provide azimuth and especially elevation errors. The errors are not too important for search radars, but may be a major effect for trackers.

6. Optical Systems

This completes the discussion of search radars. We shall now describe the manner in which infrared and optical detectors are simulated in RJARS.

Infrared (IR) detection systems in RJARS are used only in the tracking state. If they are in the connected control system, they are assigned in the same manner as radars, and go directly into the tracking state. If they are autonomous, they obtain their search information from optical systems. Hence, we shall defer discussion of the algorithms used for detection and tracking by the infrared systems to the section on tracking (UPDRT), and shall present the techniques used for optical detection.

As has been mentioned several times, the ground defensive systems in RJARS are divided into controlled and autonomous classes. All shoulder-fired IR SAMs and all freely moving guns not associated with radars are in the autonomous class. The guns include main tank guns, machine guns mounted on carriers, and fixed weapons aimed by personnel. They are designated as radar class 'G'. The optical "equipments" are persons with or without binoculars. By changing certain parameters, the optical devices could be considered to include low-level TV systems. A random draw during the data input procedure determines which individual optical systems are using magnification. The amount of magnification is a property of the equipment type, stored in RDLIB, and the random draw for each equipment is against .5, making it equally likely that magnification is or is not being employed.

6a. Detection Algorithm

At the beginning of the simulation, all optical equipments are on and in the W state. Each will scan a scenario-assigned sector of responsibility, beginning at the lower scan limit. The scan rate is such that the fixation field of view is covered in the time it takes for a successful glimpse if the contrast is sufficiently high. Typical scan rates are about 5 to 10 degrees per second. If magnification is being employed, the effective field of view is reduced, so the scan rate is reduced proportionately. The subroutine SECTSCAN moves the instantaneous direction of observation across the sector of responsibility and back at the scan rate.

The eye is treated as a contrast detector. The algorithm for determining detection capability was developed by the Night Vision Electro-Optical Laboratory (NVEOL).[2] The calculations are performed by the subroutine OPDET.

The basic concept is that the probability of detection is a function of the number of resolvable elements on the target. This in turn depends on the contrast of the target and the background, and on a reference "minimum detectible contrast". The contrast between the target and the background is taken to be the difference in brightness between the two divided by the smaller of the two brightnesses. Usually the contrast is taken as the difference in brightness divided by the background brightness. The resulting expression is then attenuated by a function of the range. With this definition, the contrast can become infinitely large, but can only become as small as -1 (perfectly black

---

[2] This algorithm has been provided to the author by RAND colleague Lloyd Mundie.

target against any background). Hence, because of attenuation, the definition strongly discriminates against the detection of dark targets against bright background, which is the usual situation with low-flying aircraft in the daytime. The algorithm may be traced to its source, the Night Vision Laboratory, who are usually interested in bright objects against dark background, the converse situation. We have chosen the unconventional definition given above, which is symmetrical between positive and negative contrast.

At present, RJARS contains no optically self-luminous objects except the flares used as IR countermeasures. All aircraft and background are illuminated only by the sun. As an automatic consequence of this assumption, no optical systems can operate at night. The flag INIGHT has been put into the scenario to provide for this condition. The aircraft are treated as two-surface objects; an upper surface directly illuminated by the sun, and a lower surface illuminated by sunlight reflected from the ground directly below the aircraft. The ground reflection coefficient is a function of terrain type and is stored in MCLUT.

The aircraft is split into three parts: fuselage, wings, and tail. The upper and lower surfaces have associated colors of paint, the particular colors being scenario input parameters PAINTU and PAINTL for each aircraft. RJARS recognizes eight colors, each designated by a two character code, as follows:

1. Aluminum	AL
2. Olive drab	OD
3. White	WH
4. Black	BK
5. Blue	BL
6. Gray	GY
7. Green	GR
0. Nonreflecting	NR

The value 0 is used for nonreflecting to designate the last color in the set. Other colors may be added, and RJARS will allocate the appropriate space. A reflection coefficient is associated with each color. The instantaneous attitude of the aircraft is used to calculate which areas are parts of the upper surface and which areas are lower surface. The tail is the same color as the upper surface.

The areas must be projected toward the line of sight. The fuselage is approximated by a cylinder, the wings and tail by planes. The fuselage is split into a nose area and a side area. The roll attitude of the aircraft determines which portion of the side area is upper surface. The tail area contributes only to the upper surface. The wings display their upper surface if the aircraft is rolled toward the line of sight, their lower surface if it is rolled away. The nose area contributes equally to upper and lower surfaces. RJARS combines all the projected areas to obtain the full projected upper and lower surfaces.

The effective contrast of the aircraft depends on whether the sun is or is not in the field of view. Observation toward the sun is more difficult than observation away from the sun. The position of the sun is found from the time of day, which is a scenario input. Due attention is paid to the latitude hemisphere to ensure that the sun appears in the proper location. The flag ISUN in the scenario establishes if the sun is visible. If the angle between the line of sight and the sun is less than 180 degrees, the sun is fully effective in reducing contrast.

The background of the target may be either sky or ground. RJARS uses the ridge files to determine this. Given the azimuth and elevation of the target, RJARS tests the line of sight to determine if it lies above the highest (furthest) ridge in the given azimuthal direction. Interpolation with respect to azimuth in the ridge file is necessary to perform this test. If the line of sight lies above the highest ridge, then the target is being viewed against a sky background.

If the line of sight is below the highest ridge, then RJARS extrapolates the line of sight from the target range to the point where it intersects the ground. The terrain at this location provides the background for the observation. It is assumed that the variation of terrain type with location is sufficiently small that a single terrain type can represent all of the terrain that is within the field of view. The reflection coefficient of the ground at this location then gives the background brightness. If we are dealing with a configuration without terrain (VIS = 0) then a smooth spherical earth is used, the line of sight is checked against the effective horizon (no refraction correction) and if the background is on the surface an average terrain reflection coefficient .1 is used.

The contrast at the target is found by combining the quantities already calculated, knowing the proportion of the target included in the upper and lower surface and weighting the reflection coefficient accordingly, to obtain the effective contrast. There are many details involving presence or absence of the sun in the field of view, which puts factors of 5 in various elements of the expression. The resulting contrast must then be multiplied by a contrast attenuation factor to find the contrast at the viewer. This factor depends on the atmospheric extinction coefficient, which is a function of the season, the location, and the seeing conditions. It also depends on a parameter called the sky-ground ratio, although it really has nothing to do with the physical sky-ground ratio within the field of view. The terminology is embedded in optical history. An examination of the data on these parameters showed that at least for European locations the values depended slightly on location, so that dependence has been dropped. The seasonal dependence split into two values, designated summer and winter, although summer conditions apply essentially in April through October, and winter conditions the rest of the year. The flag ISUMMER in the scenario (1 for summer) puts the seasonal choice in the hands of the operator. Similarly, examination of data at European (specifically German) locations provided data for median seeing conditions and for conditions that are exceeded (better for the observer) only 10 percent of the time.[3] The flag ISEEING in the scenario gives this choice to the operator. The values of the parameters are in the file IRLIB.

---

[3] These data examinations were performed by RAND colleague Horace Ory.

With the choices established, the contrast at the target is attenuated with range to give the contrast at the viewer. The attenuation equation is:

$$\text{VIEWER CONTRAST} = \text{TARGET CONTRAST}/ \left[1 + \text{SKG}^* \left(\exp(\text{EXT}^*\text{RNG}) - 1\right)\right] \qquad (22)$$

The parameters SKG (sky-ground ratio) and EXT (extinction coefficient) are two by two matrices with arguments ISUMMER and ISEEING. The numerical values are stored in IRLIB.

The number of cycles (separable visual elements) across the target is proportional to the magnification, proportional to the angle subtended by the smallest dimension of the target at the viewer, and is a function of the contrast at the viewer. Analysis[4] has shown that the dependence on viewer contrast is closely approximated by a linear function of the logarithm of the contrast, so the number of cycles C is given by:

$$C = \text{MAGNIFY}^* \left(1.22^*\log(\text{abs}(\text{VIEWER CONTRAST})) + 4.09\right)^* (\text{FUSELAGE DIAMETER})/(6.08^*\text{RANGE}) \qquad (23)$$

Here MAGNIFY is the magnification of the binoculars, the fuselage diameter (feet) has been assumed to be the smallest dimension of the aircraft, and the factor 6.08 adjusts dimensions. If the number of cycles is negative, corresponding to a small region lying between plus or minus 00044 around zero contrast, the number is set to .0001. The problem is with the curve fit, and under such conditions the probability of detection should be negligible.

The detection probability is a function of the number of cycles divided by a minimum number of cycles required for detection. It may be

---

[4] Private communication from RAND colleague Howland Bailey.

split into the product of a static part and a dynamic part, where the static part is the detection probability given infinite observation time, and the dynamic part represents the effect of the finite observation time. With C the number of cycles given by the equation above, and M the minimum number, then Howland Bailey's analysis shows that the static part may be closely represented by :

$$P_s = \frac{(C/M)^{2.7 + 0.7(C/M)}}{1 + (C/M)^{2.7 + 0.7(C/M)}} \tag{24}$$

When the number of cycles is small compared to the minimum number, this is approximately $(C/M)^{2.7}$ and tends rapidly to zero. The probability is 50% at C/M = 1, 94.5% at C/M = 2, and 99.5% at C/M = 3. It has been found by extensive field testing that to detect an aircraft-type target, that is, to determine that it is present, the minimum number of cycles M should be set to 2. To recognize it, that is, to be able to observe enough features to determine what type of aircraft is present, requires M = 3.5. These values are used in RJARS for these two functions.

The dynamic part may be represented in the form

$$P_D = 1 - \exp(-(C/M)*(TIME)*(1+.45*(ANGVEL)^2)/6.8) \tag{25}$$

Here TIME is the time that the observer dwells on the target. If he is scanning, it is the glimpse time, about 1/3 second. If he is staring, it is the time he continues to observe. The variable ANGVEL is the angular velocity of the target in degrees per second. This expression comes from field tests in which the improvement in detection capability of the eye when observing a moving target was evaluated. The factor 6.8 in the exponent is an experimentally determined quantity found by the original researchers from curve fits to the data.

6b. Control of Optical Systems

With this algorithm in hand, we shall now describe how RJARS handles optically aimed weapons. Each initially scans its search sector, using M = 2. If an aircraft is in the sector, not over the horizon, and within the maximum range capability of the observer, which is taken to be that range at which the magnified target minimum dimension subtends an angle of one minute of arc, then the subroutines SECTSCAN and INBEAM are used to ascertain if the observer has scanned over the target during the current time step. If he has, the algorithm OPDET is used to determine whether detection actually takes place. It has been found by testing the algorithm that for the low-reflectance paints used on most aircraft, the detection range is quite short.

If the target is detected, the viewer goes into an acquisition phase. He or she ceases scanning, and begins to stare at the nominal target direction. The observation time in the algorithm is now the time step. At each time step, the "detection" probability is calculated with M = 3.5, corresponding to recognition. If the recognition test is successful, then the system goes into the track phase. If the viewer is controlling an IR SAM, the IR detection system is activated and it attempts to lock the IR seeker on the target. The gun is aimed and the fire control system begins its calculations. For further details see the description under UPDRT. While the system is in acquisition it can move to outside the sector of responsibility.

If the recognition test is unsuccessful, the detection probability is calculated with M = 2 to establish if the target is still being observed. If the detection test is successful, the observer continues to stare, and tests again at the next time step. If the test is unsuccessful, the system reverts to scanning, starting at the currently perceived direction of the target.

6c. Cueing by Communications Alerting

This presentation describes the operation of the optical weapons when their operators have no information about the battle configuration. RJARS provides them with broadcast information as well. If a search radar located somewhere in the field of action detects an aircraft and reports this to its command site, then the command site not only assigns a tracker from among its own resources, but begins broadcasting the detection to all the sites in the field. This has already been described. When an observer receives the broadcast information about the location of a potential target, the range and azimuth are calculated. If the azimuth is within the sector of responsibility, and the range is within the horizon and maximum detectibility limits, then the scan pattern is changed to a sawtooth centered around the target azimuth with a width determined by the library input parameter SECTWIDTH, which is stored in RDLIB in the position occupied by RDLVUL(1) for radars (optical systems are immune to antiradiation missiles, so their variables can be used for other purposes). This restricted scan width has usually been set to plus or minus degrees, narrower than the usual scan limits, but wide enough that if the aircraft turns while approaching it can still usually be found. If the system is in the acquisition phase and loses detection, but has been cued to the target, it will revert to search over the cued sector.

If the aircraft leaves the sector of responsibility while the system is in cued search, the system will revert to uncued search over its original sector, starting at its present position. If it goes into track, the observer returns to uncued search over the original sector. If the present position is within the sector, he or she begins at the present position. If it is outside, the viewer slews to the nearer sector limit and begins scanning inward.

This completes the discussion of searching radars and optical systems.

E. UPDSI--SITES AND ASSIGNMENT

Most of the material on sites and assignment is already presented in the general description, where the tests involved in assignment are listed. A number of flags are used to set the assignment conditions. The tracker assemblage is first tested to ascertain if all trackers in the system are busy, by comparing the variables RTNSI, the number of trackers at each site, and RTNSIA, the number of trackers at each site that have been activated. If no trackers are available anywhere RJARS announces the condition and exits from UPDSI. Otherwise, it selects the best data on each aircraft and performs the assignment process as described. Among the flags are:

ASFIRDET	Some radar has detected the aircraft
ASINFLD	Aircraft in the field of action
SIPSFLG, CSPSMFLG	Data on the aircraft exist at a specified site or command site
RSDEAD, RTDEAD	Search, acquisition or track radar dead (must be false)
ASRTKFLG, ASRTMFLG	Aircraft already in track at site or command site
ASRAKFLG, ASRAMFLG	Aircraft already in acquisition at site or command site
SMSTATE	State of the SAM set associated with the tracker (must be A - available)
ASHOMFLG	Aircraft on way home
INOUTFLG	Whether inbound or outbound aircraft are being tested. These two flags are used in conjunction to determine priority
RTFLG	Tracker already assigned
RSAFLG	Acquisition radar associated with tracker already assigned

When all these flags have their proper values, the numerous tests listed in the general description will be satisfied, and a tracker will be assigned to the aircraft.

During the simulation the communications links between sites and command sites may be cut or connected. The condition is marked by the flag SICOMFLG (1 = connected, 0 = isolated). If the command CUT is given to the fictitious NAC+1 aircraft in flight maneuvers (parameter X = site in question = K, see the description of flight maneuver parameters in UPDAC), then operating acquisition or tracking radars (ASRAMFLG or ASRTMFLG = 1, signifying they have been turned on by command site M) are switched to site-autonomous operation (ASRAKFLG or ASRTKFLG = 1), and the site K is not included when the command site M, to which site K had formerly reported, tries to assign a tracker to a new aircraft. The command CON reverses the process. A site can be reconnected only to its original command site. Since there is no interaction between command sites, the situation corresponds to independent battlefield commands.

As described previously, if a command site gets information on an aircraft, it broadcasts it to the field. What actually happens in the simulation is that a flag CSCOMALERT is set for that command site and that aircraft. The flag is set only if there are data at the command site about that aircraft, CSPSMFLG = 1, and the flag CSCOMALERT is not set. Thus, the flag will not be continuously reset. If the flag is newly set, the sites will be alerted to that aircraft. For those command sites that have no reporting search radars, the alerting of the site causes it to assign one of its trackers using the standard algorithm for isolated sites. The optically aimed weapons belonging to any alerted site will be cued to the aircraft. At the moment, the battlefield dimensions treated by RJARS are usually such that the information transmission via radio link is certain, although a message probability is included in the calculations. A radio propagation model may be included in RJARS in the future. The announcement of broadcasting the command alert is printed each time it occurs. When there is no information on the aircraft at the command site, CSPSMFLG = 0, and the command alert is on, it will be turned off.

As long as at least one command site is broadcasting the alert on a given aircraft, the sites will remain alerted. When finally none of them is broadcasting, or if the aircraft is shot down, the site alert will be turned off and the equipments at each site will revert to the W state for radars and uncued search for optical systems.

F. UPDRT--TRACKING RADARS

When the acquisition radar associated with the tracker has marked acquisition by achieving three successes in no more than five scans, it turns itself off and turns the tracker on in a time RTSCN. Unlike RSSCN, the scan period of a search radar, which measures duration of rotation, RTSCN corresponds to the lock-on and settling time of the tracker circuits. The tracker will then follow its target, using range and angle servos, and have both steady and random errors. It will provide position information at interpolated time steps to its associated SAM guidance system.

1. Signal and Jamming Calculations

First the jamming power is calculated. Since each tracker may be susceptible to range deception, angle deception, or both, the jamming powers in the range and angle channels are calculated separately, using the appropriate jamming effectiveness factor. The signal and signal-to-noise ratio in both channels are then determined.

RJARS has the ability to use externally generated jamming effectiveness tables to determine the susceptibility to jamming. Data have been collected for two types of tables. The first uses measured results from flight tests, and averages over all kinds of variables to find a net reduction in kill probability when a particular jammer is used against a particular radar. These data are used by setting the scenario parameter PRBTYP equal to 2. Under these circumstances, the tracker and SAM guidance system will be treated as if they were unjammed out to the point of calculation of the kill probability in the SAM endgame. The kill probability is then reduced by the factor in the table. The second table is based on measured data that give the jamming effectiveness as a function of signal-to-noise ratio. The actual data were limited in that almost all of them referred to a single value of S/N, which we treat as a threshold such that the indicated effectiveness is achieved for S/N values above that threshold. There were effectiveness values for two values of S/N for a particular jammer-radar combination, indicating an approximately linear falloff of effectiveness below the threshold, and we arbitrarily extended that result to all jammer-radar pairs for which data were collected. The second type of effectiveness table is used by setting PRBTYP equal to 3. RJARS then again treats the system as unjammed out to the endgame, then calculates the signal-to-noise ratio, determines the effectiveness, and reduces the kill probability as above.

RJARS can use either fixed or automatically adaptive servo bandwidths. If the fixed bandwidth is selected, the bandwidth in both the range and angle channels is set to 1.5 Hz, a hard-wired number that appears to be representative of many radars. If automatically adapting servos are used, characterized by setting the input parameter SRBMOD = A, then the following theory is applied.

2. Tracking Errors

The theory of error in tracking radar servos leads to the expression for the RMS error in the azimuth angle channels due to inability to follow accelerated motion (Ref. 5, p. 307):

$$\mathrm{ERR} = .4 * \mid \mathrm{ACC} \mid /(\mathrm{SRB})^2 \qquad (26)$$

Where ACC is the acceleration in angle, SRB the servo bandwidth, and ERR the error. This error is a steady lag if the motion is a constant acceleration, or is an RMS fluctuation if the acceleration is fluctuating. The error produced by noise (jamming) is (Ref. 5, p. 278):

$$ERR = .637*NHB*(SRB/(PRF*S/N))^{1/2} \tag{27}$$

where NHB is the horizontal beamwidth of the radar antenna and PRF is the pulse repetition frequency.

The combination of these two sources of error may be minimized by choosing the servo bandwidth so the derivative with respect to the servo bandwidth of the sum of the errors is zero. This optimum servo bandwidth is:

$$SRB_{opt} = [6.3*PRF*S/N]^{0.2}*[\,|ACC|/NHB\,]^{0.4} \tag{28}$$

If the bandwidth is less than 0.5 Hz, the system will be unable to follow reasonable motions of the aircraft, so the bandwidth should be cut off on the low end by 0.5 Hz. The high end of the servo bandwidth should be below the lowest vibration frequency of the antenna assembly. This vibration frequency is inversely proportional to the dimensions of the antenna, which in turn are inversely proportional to the frequency and the antenna beamwidth. An estimate of the proportionality constant yields for the maximum servo bandwidth in the azimuthal channel

$$RTMHSB = .0013*FRQ*NHB \tag{29}$$

Similar calculations apply to the elevation channel, with NHB replaced by NVB.

A tracker normally operates by measuring the error signals in the several channels, and applying corrections to drive these signals toward zero. If the errors are too large, the slope detector that provides the tracking signal will be forced out of its capability range and the tracking lock will be broken. Angle tracking circuits usually can operate in the presence of a much poorer signal-to-noise ratio than can range tracking circuits. The theory (Ref. 5, pp. 281 and 369) indicates that the integrated signal-to-noise ratio must exceed about 8 dB for the range circuit to maintain lock, but the angle circuit only requires -6 dB. Thus, the tracker can be in any of four states--range and angle, range only, angle only, or neither, depending on the jamming susceptibility. RJARS treats the four conditions separately. The time spent in each state, after the tracker is turned on, is designated by RTTIMRA (range and angle), RTTIMRNA (range, no angle), RTTIMNRA (no range, angle), and RTTIMNRNA (no range, no angle). A counter RTJMCNT determines for how many time steps any function has been jammed. This counter is used to cause an already launched SAM that is still approaching its target to lose guidance if the radar has been jammed too long. A flag RTJAMFLG prevents launch of a SAM while the tracker lock in range or angle is broken.

The time step is generally long compared to the tracking servo response time, so the errors at successive steps can be treated as uncorrelated. The subroutine SRVERR combines glint and noise errors as independent normally distributed variables, and adds the acceleration leg term as a bias error.

If both range and angle locks are broken, the condition is printed out. If only range tracking is effective, the error in range is calculated, using SRVERR without a glint term, and the signal, signal-to-noise, range, and range error are printed. If only angle tracking is operating, the errors in azimuth and elevation are calculated as independent quantities, and the signal, signal-to-noise, angles, and angular errors are printed. If all channels are working, all errors are calculated, and the signal, signal-to-noise, range, angles, range error, angle errors, and total position error are printed. Range and position errors are in feet, angle errors in degrees.

3. Conditions for Dropping Track

Tracking will be maintained until one of the several conditions for dropping track is satisfied, at which time a subroutine DROPTCK will turn the tracker off, return the tracker SAM system to the available state (unless the tracker has been killed by an ARM or the SAM launcher is out of missiles), and record the results of the tracker-aircraft interaction. The conditions for dropping track are:

1. Target shot down, either by a SAM from the launcher associated with the tracker or by a SAM from some other command site. (A command site will assign only one tracker to an aircraft; an isolated site (no communication) likewise.)
2. Tracker killed by an ARM. The ARM success probability is calculated in the section UPDAR, which follows UPDRT in the subroutine sequence, so the tracker must be turned off at the next time step.
3. Tracker reassigned. If the tracker is following a home-bound target, and it receives a command to track an inbound target, the tracker must go off so the acquisition radar can come on.
4. Launcher reloading or exhausted. If the launcher shoots at a target and misses, the tracker will remain on the target for another shot, unless the launcher has no missiles remaining. That can be a temporary condition (missiles available at the site to reload the launcher) or permanent (all missiles exhausted).
5. Target goes over horizon. This situation will have a delay of three tracker settling times, so track may be maintained during short terrain masking intervals.
6. Target goes beyond maximum range of tracker. This will occur rarely, except for very low radar cross-section targets, since the maximum range of the radar is usually much greater than the maximum range of the SAM.

7. Target not coming within critical radius. The tracker was assigned under the condition that the target was coming within the maximum range of the SAM. If, as a result of an aircraft maneuver, this is no longer true, there is no point in maintaining track.
8. Target outbound beyond the critical radius of the SAM with SAM not launched. This permits the tracker to hold the track as long as possible.
9. Target outbound while the tracker is jammed for three time steps with the SAM not launched. Because of the different range dependence of signals and jamming, if a target is outbound and achieving effective jamming, the signal-to-noise ratio will become worse as the flight continues, so the tracker will not recover and might as well quit. "Effective jamming" means that range or angle lock is broken.
10. Target outbound while the tracker is in excess clutter for three time steps with the SAM not launched. If the clutter is bad, the tracker probably cannot recover the track.
11. Tracker jammed too long with the SAM in flight. If the tracker is jammed, the SAM will fly unguided. We allow three full time steps to elapse before tracking ceases, since the system might recover, then give up. Again, being in excess clutter for three time steps will cause the tracker to break lock.

4. Interpolations and Correlated Errors

If a SAM is in flight, it requires information more frequently than once per time step. This information is supplied by a subroutine INTERPOL. A scenario parameter DTG gives the time requirement for the SAM, and the time interval DT is divided into NGUI segments, each of length (DTG/DT), which should be chosen to be an integer. We have usually set DTG = .05, which with DT = 2.5 yields NGUI = 50 segments. Values of the range, azimuth, and elevation from the tracker to the target are maintained for the present and two previous time values (they were used earlier in UPDRT to calculate the target and angle accelerations). A quadratic interpolation formula is used to obtain the range, azimuth, and elevation at the interpolatory time values, from which the true or ground reference aircraft position ACLATG, ACLONGC, ACALTG is calculated. Errors must now be added. For this short time interval, the errors in each channel are not independent but are correlated in time, so the subroutine SRVERRCOR is used for calculation. SRVERRCOR connects the past and present values of a normally distributed variable by the relation:

$$\text{ERR} = \text{PAST}*\text{COR} + \text{PRES}*\text{sqrt}(1 - \text{COR}*\text{COR})*\text{VAR2} \qquad (30)$$

where PAST and PRES are the previous and current values of the variable, COR the correlation between them (a function of the time spacing DTG), and VAR2 is one of the normally distributed zero mean unit standard deviation variables provided by NORMV.

For both glint and noise errors, the correlation is an exponential function of time. For glint (Ref. 5, p. 288), the exponent is proportional to the change in apparent angular position, which in turn is proportional to the angular rate, frequency, and target dimension, yielding in azimuth:

$$\text{COR} = \exp(-.041917*\,|\,\text{dPHI/dt}\,|\,*\text{FRQ}*\text{GLN}*\text{DTG}) \qquad (31)$$

where dPHI/dt is the angular rate in the azimuth channel, GLN the effective length of the target, and the other symbols as before. The coefficient is a complicated combination of constants and unit conversions. For noise, the exponent is the change permissible during one servo bandwidth, so

$$\text{COR} = \exp(-2*\text{PI}*\text{SRB}*\text{DTG}) \qquad (32)$$

Interpol uses SRVERRCOR to calculate the correlated errors in each channel. The glint, noise, and acceleration components are combined as independent variables to give the total error in each channel. These are added to the range, azimuth, and elevation as previously interpolated, and then these error-affected values are used to calculate the interpolated and corrupted aircraft coordinates ACLATI, ACLONGI, ACALTI. Their values are where the tracker "thinks" the aircraft is located, and they are the values furnished to the SAM guidance system.

5. Towed Decoys

As previously mentioned, the aircraft may be towing decoys used to jam trackers. There are five techniques available to a decoy. Since jamming techniques 0 (noise), 1 (search deception), 2 (range deception), 3 (angle deception), and 4 (range and angle deception) are already assigned, these techniques are designated:

5 Towed reflector. The decoy, itself small and cheap, may be carrying one or several corner reflectors that make it look like a large target. The tracker may then follow the decoy instead of the aircraft.

6 Towed noise jammer. The decoy, directed by a warning receiver, which may be on the aircraft or on the decoy itself, may radiate noise. This would be most useful against home-on-jam counter-countermeasures.

7 Towed repeater. The decoy amplifies and repeats all tracker signals it receives in its coverage band, providing the tracker with a strong false target. For those trackers that follow the center of radiation of the received signal, the information transmitted to a command guided missile will be erroneous.

8 Blinking noise. The aircraft and the decoy alternately radiate noise. The idea is to make the tracker oscillate between the two targets, with a resultant loss in accuracy.

9 Blinking repeaters. Like technique 8, but the reradiated signal is an amplified repeated signal. It is expected to be more effective than 8.

The blinking techniques can be employed by two independent aircraft. The commands BLINK and BLKPR in the aircraft command list (see UPDAC) make the aircraft partners and allow them to use these techniques.[5]

6. Semiactive Systems and Illuminators

The SAMs associated with tracking radars are of two types, command guided and semiactive. The command guided SAMs use information directly provided by the tracker for their guidance systems. Semiactive SAM systems have an illuminator associated with the tracker. The illuminator is slaved to the tracker, and radiates a signal, usually either CW or pulse doppler, which is reflected from the target and received by the SAM in flight. The seeker on the SAM is a monopulse receiver which uses servos and points itself at the center of radiation of the reflected signal. The details will be presented in UPDSM. Such receivers are susceptible to the decoy techniques described above.

The illuminator is turned on at the time of launch. If the target is distant or of low radar cross-section and the illuminator is weak, which is a quite common situation, the received signal at the seeker may be below the threshold value which the seeker requires to operate. What RJARS does is to provide the semiactive seeker information at launch on the aircraft current position and velocity vector. If the seeker signal is adequate, the seeker behaves normally. If the signal is below threshold, the seeker calculates the projected position of the target,

---

[5] It is the author's opinion that the blinking techniques are less effective against seekers than the decoy-only radiating techniques. With blinking, the seeker will swing from one target to the other. When the SAM is sufficiently close, one of the targets will go out of the field of view and the seeker will lock on the other. If the final target is the decoy, great, but if it is the aircraft, curtains. Since there is no relation between the time phase of the blinking cycle and the distance of final lock on, the chances are 50 percent that the blink cycle will be in the wrong phase. Hence, the author believes that the blinking techniques 8 and 9 should not be used against illuminators.

assuming no maneuvers, and guides toward that point. If eventually the signal becomes strong enough, the normal guidance system takes over.

The main advantage of a semiactive system is that the tracker need only point toward the target with sufficient accuracy to keep the target within the illuminator beam. Semiactive systems are very well suited for track-while-scan operation, in which one tracker aims several illuminators. This has not been implemented in RJARS, but it may be in the future. Also, the accuracy of the missile does not degrade as badly with range as it does for command guided systems. The angular accuracy of a tracker will fall as the square of the range (proportional to the square root of the received signal power, two-way propagation) while the accuracy of a seeker will be almost independent of range (short distance from seeker to target during final approach providing strong signal). Consequently, many modern systems use the semiactive approach.

Clutter and multipath effects are included for trackers. The calculations are identical to those for search radars. However, the effects of clutter and multipath are more severe for trackers. Lock may be broken by clutter, or the errors may become large. If the multipath effect is strong and the phase relations are right, the tracker will point at the ground below the target (the centroid of the target and its image). RJARS follows ESAMS in making detailed calculations of the location of the specular point and the amplitude and phase of the reflected signal. A separate random number generator NORMC is used so the random number sequence for the main simulation will be the same in the presence or absence of clutter and multipath. The multipath effect on the signal amplitude and the elevation error are worked out in detail.

Trackers have procedures to reduce the effects of clutter. Following ESAMS, these techniques are subsumed in a single expression TRKSCVD (tracker subclutter visibility in dB), an input number stored for each tracker type in RDLIB. After the clutter signal (not the multipath) has been calculated, it is reduced by the appropriate factor. Generally, pulse doppler radars have better clutter reduction than ordinary pulse radars.

7. Infrared Systems

We shall next describe the methods used by IR systems in RJARS. The assignment of IR trackers, either centrally controlled or autonomous, has already been discussed, so it will be assumed that the IR equipment is in the tracking state (RDSTATE = 'Q'). The algorithms used to determine signal strength will be presented. All the numerical data used for IR systems (and also for optical systems) are in the file IRLIB. Most of the IR material has been derived from ESAMS.

An IR tracker detects the thermal radiation from a target, finds the direction of the center of radiation, and uses the resulting angular information to guide its missile. The tracker is located on the missile, and we shall consistently refer to it as a seeker. To develop the theory of IR seekers we need to know the properties of the radiation from the source.

7a. Source Radiation and Attenuation

Following ESAMS, we assume that the seeker works in either of two spectral bands, and measures the radiation from the source integrated over the appropriate band. The radiation from each type of aircraft is presented in tables similar to the radar cross-section tables. For each band, the radiant output from the aircraft in watts per unit solid angle (watts/ster) is given as a function of the azimuth and elevation angles of the line of sight to the aircraft. Azimuth is measured in the horizontal plane of the aircraft, with zero at the tail, 180 degrees at the nose, and right-left symmetry assumed. Elevation is measured from directly below (0) through the horizontal plane (90) to directly above (180). The tables are in steps of 15 degrees.

This procedure regards the aircraft as a single source of radiation, with a directivity determined by the shadowing of the hot parts around the engines by the body structure. The parts of different temperature radiate different amounts of energy in the two wavelength bands, so the directionality is different. We shall shortly say more about the separation of the radiation into body and jet plume radiation.

The radiation at the source will be attenuated as it travels through the atmosphere. The attenuation is a function of wavelength, so the proper procedure is to take the radiant source energy in each direction as a function of wavelength, multiply by the attenuation factor at that wavelength (the exponential of the product of the attenuation constant and the range), then integrate over the wavelength coverage at the seeker. This is sufficiently complex that it requires an off-line calculation. The results for ESAMS were developed using an unspecified computer program. Some further calculations at RAND used the LOWTRAN program.

For proper representation of the attenuation, it should be presented for each aircraft type as a function of type, wavelength, range, azimuth, and elevation. ESAMS has simplified this situation by using the same attenuation for all aircraft types. Thus, the attenuation data in IRLIB are presented as ten tables. For the two bands and for five ranges (1000, 5000, 13000, 30000, and 99000 feet), the tables give the attenuation for elevation angles between 0 and 180 degrees, and for azimuth angles between 0 and 90 degrees (tail to beam). It is assumed that the engine radiation is sufficiently shielded in the forward hemisphere that only the body is radiating, so the attenuation in the forward hemisphere is taken as the same as in the beam plane.

The effect of this approximation is that all aircraft are treated as having the same wavelength-dependent shielding, in the sense that the spectral distribution of the radiation in a given direction is taken to be the same for all aircraft. This is clearly a questionable assumption. For example, an aircraft with engines above the wing, such as the A10, will put less engine radiation into the lower hemisphere than will an aircraft with engines below the wing. Since the engine radiation is hotter than the body radiation, more of it will lie in the shorter wavelength band. Since the spectral distribution in the bands is different for the two types of aircraft, the attenuation will be different, and this should show in the tables. However, lack of data on the variation of spectral distribution with type has caused us to retain the ESAMS procedure.

RAND colleagues Lloyd Mundie and Horace Ory studied this effect. For two fixed-wing aircraft types (A10, F16), and three helicopters (AH64, OH58, V22), the radiation and its spectral distribution were calculated as a function of observation direction. The LOWTRAN program was used to obtain the attenuation for each type as described above. The original radiation was split into the body (including engines) and the jet plume, and separate calculations were performed for each. These improved data, and any that may result from further calculations, will be incorporated into RJARS in the future, with a flag to show for any aircraft type whether it uses the original calculations or the new and improved variety.

7b. Signal Calculations

The detailed calculations, performed by the subroutine IRSIG, determine the angles between the line of sight and the oriented aircraft. Interpolation in the radiation tables gives the radiant output of the aircraft in the proper band in the indicated direction. The range is determined (this information is not known to the actual seeker, but it exists in reality, the program knows it, and can use it to perform the calculations) and three-variable interpolation in the attenuation tables gives the attenuation factor. The product of the radiant intensity and the attenuation, divided by the square of the range, gives the signal received at the seeker. When the Mundie-Ory model is incorporated, the signal from the body and the plume will be found separately and added.

In the daytime, the only relevant time for IR operation, the aircraft will be illuminated by the sun and the seeker will receive the reflected sunlight. The reflected solar energy is treated as a fixed constant for each band, multiplied by the attenuation factor of solar radiation in that band. Many seekers use detectors that work in an IR wavelength band where solar radiation is negligible. The solar signal is added to the self-produced signal to give the total signal to the seeker.

The signal is divided by the equivalent noise of the seeker to obtain the signal-to-noise ratio. The equivalent noise of an IR system is subject to several mutually inconsistent definitions. The one used in ESAMS and RJARS is that the equivalent noise is the sensitivity of the detector in watts/sqm, a property of the material, multiplied by the detector area, divided by the area of the collecting mirror, and divided by the transmissivity of the seeker optics. It thus is the sensitivity of the detector divided by the optical gain of the system, and is the minimum detectible signal at the input to the optics. The signal calculated above in watts/sqm is at the input to the optics, so the calculation is consistent.

For the system to operate properly, the signal-to-noise ratio must exceed a threshold value. ESAMS and RJARS generally set this value to 2. At this point, an approximation to the effects of background clutter is introduced. If the field of view of the seeker when pointing at the target includes the ground, the threshold is increased from 2 to 7. A correspondence with BDM Corporation, the developers of ESAMS, revealed that this approximation was provided many years ago by the Army Helicopter Group in St. Louis. Discussions with RAND's specialists in IR systems indicate that not enough is known about the effects of IR ground clutter to warrant further investigation.

When the IR seeker is initially turned on, it begins to calculate the received signal. When the signal-to-noise ratio exceeds the clutter-modified threshold, the integer SKLOCK is set to 2 and the launch procedures proceed exactly as for radar equipments. If the signal-to-noise ratio drops below threshold, SKLOCK is decremented. If SKLOCK drops to 0, meaning loss of signal for two time steps, the seeker will drop track. If the SAM has not been launched, the seeker will continue looking for the target. If it has been launched, self-destruction ensues.

IR seekers can be decoyed by flares. RJARS treats this possibility, launching and flying the flares and finding the resultant combined signal and angular displacements. The details are in UPDSM.

G. UPDAR--ANTI-RADIATION MISSILES

As mentioned in the general description, each aircraft may carry several anti-radiation missiles. Each missile has an associated frequency band, corresponding to its receiver coverage. ARMs may have scenario-specified targets, corresponding to known radar locations, or may shoot at targets of opportunity. UPDAR controls the assignment, flight, and results for the ARMS. There is no interaction between ARMs and IR or optical systems.

UPDAR first checks for each aircraft that it is in the field of action, has missiles on board, and either the aircraft is still alive or ARMs it has launched are still flying. It then checks the radars to verify if they are turned on and if they have been registered by the warning receiver on the aircraft. A flag ARMLNCH ascertains whether an ARM has already been launched from that aircraft at that radar. Only those acquisition or tracking radars that are pointed at the aircraft are appropriate for ARM attack.

If no ARM has been launched at a radar, UPDAR determines whether one should be launched. It tests that the aircraft is approaching the radar, and then checks for each missile still on board that the radar range is between the ARM's maximum and minimum flight ranges, that the radar frequency is within the frequency band of the ARM, and that the ARM is either preprogrammed for that radar or is opportunistic. If all these conditions are satisfied, an ARM is launched.

The counter ARMNL tells the aircraft how many ARMs it has left. If ARMNL = 0, no further ARMs may be launched by that aircraft. Since the ARMs have self-contained guidance, they can continue flight even if the parent aircraft is shot down, at least until the target radar ceases transmitting.

The ARM trajectory used in RJARS is much simplified from what we believe to be actual ARM trajectories. The ARM flies at constant velocity, rather than initially accelerating away from the aircraft and then slowing from drag. The ARM velocity used in RJARS is the velocity with respect to the ground, and may be thought of as the sum of the velocity of the launching aircraft (assumed to be aimed at the target radar) and the velocity imparted to the ARM by its propulsion system. Aircraft aiming maneuvers are not included in RJARS.

The ARM is launched on an initially lofted trajectory on the bearing connecting the aircraft and the target. It goes into a 1-g pitch-down maneuver until the pitch of the vector velocity of the ARM drops below the line of sight from the ARM to the target radar. The pitch is then maintained along the line of sight, so the trajectory is straight. The position of the ARM and its range to its target are printed out at each time step during flight.

Anti-radiation missiles require a radar signal for their guidance system operation. If the target radar goes off the air, which may occur for the variety of reasons presented in the sections on radars, then an ARM used in RJARS will lose guidance. It will continue to fly for a time ARMTDLY, a library input, with the kill probability reducing linearly to zero at the end of the no-guidance interval. The alternative possibility that the ARM can divert to another target has not been implemented except as described in the next paragraph. This is a limitation of the simulation.

In normal operation, if an acquisition radar locks on to its target, it turns its tracker on and turns itself off. It is possible that an ARM was in flight against that acquisition radar, especially if the radar is being jammed. If the acquisition radar and its tracker are in close proximity (the hard-wired value is .01 nmi = 60 ft) and both the acquisition radar and the tracker are in the frequency coverage of the ARM, then the ARM will transfer its attention to the tracker. Conversely, if the ARM was directed against a tracker which is in autonomous operation and reverts from track to acquisition or search, the ARM will continue to fly against that tracker if the acquisition or search frequency is in the coverage band of the ARM.

The arrival of an ARM at its target is marked by the ARM altitude dropping below the radar altitude. The time of arrival is then determined by interpolating the linear trajectory back to the radar altitude. The vulnerability of a radar type to an ARM type is a parameter array RDVUL (five elements per radar, corresponding to five

ARM types, which was stored during the data read-in process). This vulnerability is compared to a random draw and the outcome of the ARM's attack on the radar thereby determined and printed. If the attack is successful, the flag RSDEAD (search radars) or RTDEAD (track radars) is set equal to 1, and that radar will no longer participate in the simulation.

H. UPDSM--SURFACE-TO-AIR MISSILES

Much of the theory and programming for surface-to-air missiles has been derived from the computer program ESAMS (Ref. 6).

After the tracker has established lock-on to the target aircraft, its associated SAM system must be alerted, counted down, the SAM launched, flown under its propulsion system, guided to the target, and a mission outcome determined. These procedures are implemented in UPDSM. An option parameter, NOSHT, will bypass UPDSM if it is set equal to Y. This option may be used if it is desired to study only the target-radar-jamming interactions.

1. Sequence of Operations

The condition of a SAM is characterized by the parameter SMSTATE, which has the following possible values.

A	Available
P	Preparing
F	Ready refire
R	Reloading
E	Exhausted
B	Boosting
I	Interstage
S	Sustaining
C	Coasting

The "SAM" may be an anti-aircraft artillery weapon. The identification is determined by the parameter SMGUICL, which has the values:

A	Artillery
C	Command guided
S	Semiactive
I	Infrared

An unassigned SAM is in the A state. If it is determined that it should be launched, it goes into the P state for countdown. If it is launched successfully, it steps through B, I, S and C in the flight. After the attack endgame, the number of missiles at the launcher is checked. If they are all gone, but there are still missiles in storage at the launch site, the system reloads in state R. If all missiles at the launch site have been expended, the launcher falls into state E and leaves the simulation. If there still are missiles at the launcher, then if the attack was successful, the system reverts to state A for further assignment. If the attack was unsuccessful, the system enters state F, where it remains assigned to the current target, but the reaction time is reduced from that of state A.

The reaction and refire times in RJARS are drawn from distributions. Either uniform or normal distributions may be employed, the control being the character flag RCTDISTRIB in the scenario, which may be set to U or N. For each SAM type, two numbers are stored for reaction time and for refire time, representing the upper and lower limits for each. A uniform distribution draws between these limits, a normal distribution uses the average of the two for its mean and 1/6 of the difference for the standard deviation, with the distribution cut at the limits (3 σ points). It is expected that lognormal distributions for reaction and refire times will be added to RJARS in the future.

Only SAMs in state A or state F may be launched. If the tracker has been tracking its target in both range and angle for more than two time steps, then the call goes to the SAM system to execute the subroutine LNCHCALC. In this subroutine, the flight of the SAM is approximated as a constant velocity flight after booster cutoff. The aircraft path is projected with its present velocity in its present direction. The SAM path is drawn to intersect the aircraft path, and the pitch and heading of the SAM are adjusted so that when the paths cross, both the SAM and the aircraft are at the point of intersection. The time until intersection includes the countdown delay time, during which the SAM is stationary but the aircraft is flying. When the range from the SAM to the radar is equated to the range from the aircraft to the radar, there results a quadratic equation in the time. This equation is solved by LNCHCALC to obtain both time and range of intersection. If the resulting range lies between the maximum and minimum range capability of the SAM (library inputs), then the SAM enters state P and a nominal launch time is set at the countdown duration from the present (upward adjusted to an exact time step).

If the "SAM" is an anti-aircraft artillery weapon, a different procedure must be followed. The approximation of constant velocity is not appropriate for a shell, which starts at high velocity and slows down from the effects of drag and gravity. The shell trajectory is approximated as a constant drag coefficient flight, so the velocity is an inverse function of time, and the range is a logarithmic function of time. This trajectory is employed to find the point and time of intersection, the launch elevation, and the launch heading. These values are then corrected to first order for the effects of variable density and gravity drop.

When the nominal launch time is reached, the current situation is checked. The aircraft may have maneuvered so that if its position, projected to the time of SAM booster cutoff, is either closer than the range of booster cutoff or beyond the maximum range capability of the SAM. Also, it is possible that the tracker is now being jammed. If these events occur, the SAM launch is delayed till the next time step. If not, the launch command is given.

The SAM launch reliability is compared to a random draw. If there is a failure, the system transits to the F state and begins another countdown. Otherwise, the SAM launch parameters are initialized. The time and range of the expected intercept point are found from LNCHCALC.

Projecting the aircraft forward by this time yields the coordinates of the projected intercept point, from which the SAM launch azimuth and elevation are found. The SAM elevation is incremented by a loft angle. The aircraft may be in a dive at present, which would cause the projected intercept to be underground. If so, the launch elevation is set to the loft angle. The SAM enters the B state, a launch flag SMLNCHD is set, and the success or failure is printed.

If the launcher is an optically aimed gun, the accuracy of aiming is simulated by the parameter SKAIMERR, stored in RDLIB. Independent normally distributed errors with mean zero and rms value SKAIMERR are added to the initial aiming direction of the gun. For most guns, this error is more important than the dispersion in the launch direction.

2. SAM Propulsion and Guidance

The SAM must now be flown out to intercept. As discussed in the general description, the time step must be shortened if any accuracy of flight is to be achieved. The propulsion and guidance system used in RJARS is a hybrid system in which propulsion is applied along the body axis of the SAM, and the heading and pitch are steered directly from the tracker or seeker with a lag to approximate the missile dynamics.

The SAM propulsion system is contained in the subroutine SAMACC. This subroutine implements the equation

$$\mathrm{ACCEL} = (\mathrm{THRUST} - \mathrm{DRAG})/\mathrm{MASS} \tag{33}$$

All SAMs are modeled as four-stage devices--an unguided booster, an interstage coast which may have zero duration, a guided propelled sustainer, and a guided unpropelled coast stage. In each stage, the thrust is constant (zero for the coast stages), and the mass decreases linearly with time. The thrust, mass at the beginning of the stage, and mass rate are library inputs. All the SAMs on which we have collected data have used metric units, so the thrust is in Newtons, the mass in kilograms, the mass rate in kilograms/sec. Factors of MTF (meters to feet) appear in the equations to convert the acceleration to the English units used elsewhere in RJARS.

The drag is a function of the atmospheric density and the Mach number. A subroutine ATMOS (copied from ESAMS) calculates the atmospheric density and the velocity of sound as functions of the SAM altitude. The Mach number is the ratio of the SAM velocity to the velocity of sound. The drag force is:

$$DRAG = .5*DENSITY*(REF.AREA)*(VEL/MTF)^2*CD \qquad (34)$$

where the reference area is in square meters, the SAM velocity in fps, and CD denotes the drag coefficient.

For an actual SAM, the drag coefficient is a function of Mach number, the angle of attack of the SAM with respect to its flight path, and the position of the guiding fins. The latter two variables are not included in RJARS, so the drag coefficient is treated as a function of Mach number only.

For almost any SAM, the drag coefficient will be approximately constant at low velocity, will increase rapidly as the velocity of sound is approached to a maximum for M (the Mach number) slightly beyond unity, then will decrease again as the airflow around the SAM becomes smoothly supersonic. The drag coefficients for the three stages will have approximately the same shape versus M but will have different numerical values. We have found by studying an ensemble of SAMs that the drag coefficient is well fitted by a set of inverse quadratic functions of M. Let the three stages be denoted 1 (boost), 2 (sustain), 3 (coast). The interstage coast uses the drag of the sustainer. For each stage, define three transition values, designated MACHT in the program, thereby forming a three-by-three array MACHT (K,L) where K denotes the stage. Then the drag coefficient is given by:

$$CD = CD1(K,L)/(1 + CD2(K,L)*M^2) \quad (35)$$

If the Mach number is below MACHT(K,1), L = 1 is used in Eq. (35), if it is between MACHT(K,1) and MACHT(K,2) L = 2, and if it is between MACHT(K,2) and MACHT(K,3), L = 3. For Mach numbers above MACHT(K,3), set the Mach number equal to MACHT(K,3), thereby keeping the drag coefficient constant at high velocities.

The coefficient CD1 is always positive. For the booster, CD2(1,1) = 0 or a small negative value, corresponding to the near constancy of the low velocity drag coefficient. CD2(1,2) will be negative and is usually between -.4 and -.6. All others coefficients will be positive. Actual calculation of the drag coefficient parameters may be done quite readily from a curve of drag coefficient versus Mach number; the curve is generally provided with SAM data.

The SAM velocity and position are updated with a second-order Runge-Kutta equation. The subroutine SAMACC, which takes as inputs the SAM altitude and velocity, is used with their values at the previous subdivided time step to calculate the acceleration at the beginning of the last interval. The position is updated using the past velocity, pitch, and heading. The velocity is then incremented with the calculated acceleration, and the mass is decremented. The new altitude, velocity, and mass are entered into SAMACC to obtain the acceleration at the end of the interval. The velocity and position are then corrected, yielding as final results the equivalent of allowing the acceleration to vary linearly over the subdivided time interval.

The direction and range from the SAM to the target are calculated. During the boost stage, no guidance is applied to the SAM, so its heading is maintained constant. The pitch decreases steadily, following the loft program. During sustain and coast, for command guidance, the aircraft and SAM paths are projected forward along their current headings to their point of closest approach. The azimuth and elevation of the line of sight from the SAM to the position of the aircraft at its time of closest approach then determine the direction toward which the

SAM should be pointed. Semiactive and IR SAMs use proportional navigation.

If the "SAM" is an artillery shell, the flight is unguided. The acceleration and velocity are calculated as above. The pitch is corrected for the effects of gravity drop, and the heading is maintained at its launch value. The position is then updated.

Let S denote the vector from the SAM to the aircraft, with present value S_0, and let $u = v_M - v_A$ denote the relative velocity of the aircraft and the SAM. Then the magnitude of the square of the vector S is given as a function of time from the present by

$$S^2 = S_0^2 - 2(\underline{S_0}\cdot\underline{u})*t + u^2*t^2 \qquad (36)$$

This has a minimum at the time

$$t_1 = (\underline{S_0}\cdot\underline{u})/u^2 \qquad (37)$$

at which the position of the aircraft with respect to the ground is given by:

$$\underline{x}_{A1} = \underline{x}_{A0} + \underline{v}_0*(\underline{S_0}\cdot\underline{u})/u^2 \qquad (38)$$

The SAM velocity vector is commanded to point along the vector from the SAM to the position X_{A1}. However, a true SAM could not execute such a command, which requires instantaneous velocity changes, since it can only apply finite accelerations, using its fins or thrust vector controls. To simulate the situation without requiring excessive complication the dynamic response of the SAM has been approximated by a simple lag response, characterized by the library input TIMECON. Thus, the SAM heading H responds to the commanded direction A (the azimuth of the line to the projected point described above) by the differential equation

$$(\mathrm{TIMECON})*dH/dt + H = A \tag{39}$$

The direction A is assumed to vary linearly with time from the beginning to the end of the subdivided time step. With this approximation, the solution of Eq. (39) expresses the heading at the end of the interval in terms of the heading at the beginning of the interval, the commanded azimuth at beginning and end, and three lag parameters:

$$\begin{aligned} H(t + \mathrm{DTG}) = {} & H(t)*\mathrm{LAGCON}(1) + A(t)*\mathrm{LAGCON}(2) \\ & + A(t + \mathrm{DTG})*\mathrm{LAGCON}(3) \end{aligned} \tag{40}$$

where the three lag parameters, calculated during data input, are given in terms of the time constant by:

$$X = \mathrm{DTG}/\mathrm{TIMECON} \tag{41a}$$

$$\mathrm{LAGCON}(1) = \exp(-X) \tag{41b}$$

$$\mathrm{LAGCON}(2) = (1 - \mathrm{EXP}(-X))/X \; - \; \mathrm{EXP}(-X) \tag{41c}$$

$$\mathrm{LAGCON}(3) = 1 - (1 - \exp(-X))/X \tag{41d}$$

This guidance system is a modified form of lead pursuit guidance. The information about the aircraft position comes from the tracking system in terms of the variables ACLATI, ACLONGI, ACALTI, so it is already corrupted with tracking errors. The SAM will stage itself at booster cutoff (enter state S - sustaining) and at sustainer cutoff (enter state C - coasting). If the flight time exceeds the maximum flight time available to the SAM, it will self-destruct and so announce.

Semiactive and IR SAMs employ proportional navigation, for which the commanded rate of change of heading is proportional to the angular change of the line of sight from missile to target. The proportionality constant, SMGUIGAN, is stored in SMLIB for each missile type. The equations may be integrated once, yielding the change in heading during a subdivided time step in terms of the commanded change and the SAM lag constants. Because of the higher order equations, an additional lag constant appears in the expressions.

Proportional navigation systems are well known to be subject to instability in the final approach stage if the gain is too high. The problem is alleviated by making the gain a function of the time to go. This has not been implemented in RJARS, but if the closing velocity is high, the effect should not be too important.

3. Semiactive Seekers

The semiactive and IR seekers are updated in UPDSM. The semiactive seeker is a monopulse radar receiver. Briefly, it is simulated as a four antenna system. One pair of antennas is displaced in the horizontal plane, the other in the vertical. Antenna patterns are calculated using the subroutines SEKRPAT, which finds the pattern of an individual antenna (taken to be the pattern of a horn antenna rather than a paraboloid), and MONOPAT, which assembles the patterns of the four oriented and displaced antennas. The sum of the four outputs (SKSUMSIG) is used to find the total returned signal. The combined signal from the two right displaced antennas is compared to the signal from the two left antennas. If it is greater, then the target should be to the right of the antenna vertical midline. Similarly, the signal from the upper antennas is compared to the lower pair to find if the target should be above or below the horizontal midline. These signals, designated SKDIFFSIG[0] and SKDIFFSIG[1], are divided by the sum signal and properly scaled to provide azimuth and elevation input commands for the guidance system. The scaling is established in the opening GETDATA section of RJARS by finding the values of command signal produced by .01 degree displacements. The subroutines ORIENT and REORIENT respectively convert the angles from inertial space to the nominal sight line from target to missile and back. Monopulse seekers are generally limited to about a beamwidth of coverage, since outside that value the correction signal has the wrong sign. This effect is included in RJARS. The beamwidth is that of the seeker and may be considerably wider than that of the tracker.

3a. Seeker Clutter

RJARS first calculates the signals as if the seeker were aimed perfectly at the target. It then looks at the effects of clutter. Clutter in a pulse doppler or CW system is more complicated than is the clutter treatment in a pulse radar system. The procedures used to treat clutter follow those of ESAMS. For clutter to affect a pulse doppler system mounted on the seeker, the following conditions must be satisfied:

1. The clutter signal must lie within the range gate of the seeker. Thus, the path length from tracker to clutter region to missile may differ from the path length from tracker to target to missile by less than the range gate width. Since a constant value for the sum of two such lengths corresponds to an ellipse, this condition defines a pair of ellipses on the ground.
2. The clutter signal must lie within the doppler filter width around the signal doppler frequency. The doppler frequency of the clutter signal is determined by the aircraft and SAM velocity, which set the rate of change of the two path lengths. Since the two lengths involve the coordinates of the ground point with opposite signs, there results a condition on the difference of the two path lengths. A constant value for the difference of two such lengths corresponds to a hyperbola, so the ground point must lie between two hyperbolas.
3. The clutter signal will lie in the doppler filter band if the doppler frequency produced by the path difference differs from the target doppler frequency by a multiple of the pulse repetition frequency. Consequently, there will be a set of paired hyperbolas on the ground which can produce clutter

signals. These may extend out to the horizon of mutual visibility.

4. The range gate width and doppler filter width are generally small enough that the ground properties do not change appreciably over the ground regions allowed by conditions 1 and 2. Thus, two ground points can be located, determined by the possible intersections of the ellipse and hyperbola at the center of the regions. The effective area of each ground region can be calculated analytically by approximating it as a parallelogram.
5. The ground points must now be tested for visibility to the tracker, the missile, and the target. Since the position of the missile is not known prior to the simulation, this calculation cannot be performed by the preprocessors, but must be made at each subdivided time step during the missile flight. The subroutine MASK performs the detailed calculations, which are essentially the same as those described earlier in UPDTR.

If the ground points are visible to both the seeker and the target, the seeker antenna gains in the proper directions are used to find the clutter signal from each point. The sum and difference signals of the clutter are combined with those of the target to find the net input signal to the guidance system. The appropriately weighted signals from all visible hyperbola pairs must be included. While seeker clutter occurs rather infrequently, because of the severity of the conditions above, when it does occur the effects may be quite severe and cause the seeker to lose lock on the target.

Clutter in a CW seeker is a somewhat nastier problem. There is no range gate, so the ellipses do not limit the clutter region. The clutter must be integrated over the region between each pair of hyperbolas, out to the visibility limits. Each ground point between the hyperbolas must be weighted by its area, which involves the distance from the midline of the hyperbola (the angle bisecting the asymptotes), and by the antenna pattern in the proper direction. This calculation is quite complex, and significantly slows the RJARS operating speed. At the moment, CW clutter is commented out of RJARS. When the theory is improved the effect will be restored.

While clutter effects on seekers are quite complex, the effect of multipath should be negligible. For a seeker to be affected by multipath, not only must the return meet the conditions above, but the ground regions must include specular reflection points. The probability of this occurring is sufficiently low that seeker multipath effects are not included in RJARS.

3b. Jamming of Monopulse Seekers

Jamming effects on monopulse seekers are considered. The usual technique for jamming a pulse doppler seeker is velocity gate pull off, in which the jammer generates a signal at a deceptive doppler frequency, varies it in time to cause the velocity gate to move away from the target doppler velocity, then turns off the deception signal to leave the seeker hanging in limbo. A more sophisticated version varies the timing of the deception signal so the range gate is also deceived. RJARS does not handle this procedure, but includes it under range and angle deception of the tracker. If desired, simulation of velocity gate pull off is fairly simple.

The effect of decoy deception of seekers is included. When a decoy is present, the combination of target and decoy signals, weighted according to their relative strength, is used to get the net sum and difference signals for guidance input. The reflected and radiated signals are included as the variables SKINTERF[0] and SKINTERF[1]. The effect is to cause the seeker to follow the apparent center of radiation, which, it is hoped, lies close to the decoy.

The target and decoy signals are proportional to the antenna patterns in their respective directions. Since only the motion of the seeker direction is controlled, these patterns must be evaluated from the predicted position of the seeker from its last position and the seeker servo equations. This effect can tend to make the seeker wander between the two sources rather than lock on one of them.

As the missile approaches its target, the angular separation between the target and the decoy increases until only one of them is in the seeker field of view. At that point, the seeker should snap to the direction of the remaining signal, limited by its rate of servo motion. For typical seeker beamwidths and decoy separations this event will usually not occur until the last 1000 to 2000 feet of travel.

4. Infrared Seekers

Infrared guided SAMs operate very similarly to semiactive SAMs. The launch procedure requires that the seeker actually be locked onto the target. Since IR SAMs are generally pointed by optical equipments, the relative sensitivity of the IR and optical determines whether the SAM has to wait a long time after being aimed until the seeker locks on. Once the SAM takes off, it uses the onboard seeker to provide the angular information for the proportional navigation guidance system. The maximum angular rate capability of the seeker is a function of the received signal level. For signals below the minimum level, the seeker will not move. Above the upper critical level, the seeker can move at its maximum rate. In between, the seeker rate is limited to an amount proportional to the signal level. The critical levels and maximum rate are stored in SMLIB.

The rate of response of a seeker is usually very fast, so it moves to the center of radiation of the received IR energy. The resulting pointing direction is transmitted to the guidance system. Such a system clearly can be decoyed by flares. RJARS has adapted the ESAMS flare treatment.

4a. Flares

Only a single type of flare is provided in RJARS. This may be changed if data on other flare types become available. Each airplane can have zero, one, or two flare dispensers, the number and dispensing pattern being governed by the variable ACFLPAT for each aircraft in the scenario. The position of each flare dispenser is displaced longitudinally along the fuselage midline from the aircraft center, and laterally perpendicular to the midline. A single dispenser is along the midline. Dispensers are tilted with respect to the vertical, and pointed in an arbitrary azimuth direction. If there are two dispensers, they have the same tilt from the vertical, and symmetric azimuth. The number of flares in each dispenser is specified in the variable ASLFLARE0, stored in IRLIB for each aircraft type (same number of flares in each dispenser).

Flares are dispensed with a constant time separation. Two dispensers may fire simultaneously or alternately. If the latter, a random draw at the beginning of the simulation determines which dispenser fires first.

The data on flares are stored in IRLIB. The flare brightness is given as a lookup table against time and flare velocity. Its mass and drag coefficient are given as a separate lookup table using the same variables but with different values for the time and velocity arguments. Flare radiation is treated as omnidirectional, and the attenuation factor for the two bands is stored for the same values of range as used for the *aircraft attenuation data*.

The decision to launch flares in RJARS is reactive. Two numbers are stored for each aircraft in IRLIB, representing the probability of detecting a missile launch from nose-on or tail-on incidence. The nose-on probability is derived from discussion with pilots resident at RAND, the tail-on probability from some consideration of the capabilities of IR missile warning detectors. The detection probability is fitted with a cosine function in azimuth between the nose and tail values, and a cosine of the elevation angle to partially account for the obscuration of the line of sight. When the IR missile is launched, a random draw determines if the aircraft is successful in detecting the launch, and the result is printed.

If the detection is successful, the aircraft begins dispensing flares according to its specified pattern. The probability of successfully launching each is determined by a random draw against IRFLREL, the reliability of launching and lighting the flare. If successful, the flare is released with an initial velocity with respect to the aircraft IRFLVEL, in the direction set by the tilt and azimuth of the dispenser. These variables, plus the time separation of flares IRFLINT, are stored in IRLIB. The flare is flown by the subroutine FLYFLARE, which operates every subdivided time step DTG. The "drag coefficient" stored versus time and velocity is actually the product of the true drag coefficient and the area of the flare. FLYFLARE flies the flare like an unguided SAM, interpolating in the lookup tables for mass and drag coefficient and updating the velocity and position. The characteristics of flares are usually such that they decelerate rapidly and fall behind the aircraft. If the tilt and azimuth angles are properly chosen, forward flying flares can be simulated.

4b. Signal from Target and Flares

The IR radiation in each spectral band is calculated from the lookup tables knowing the flight time and the velocity. This gives the signal at the seeker received from each flare. RJARS contains a rudimentary scheme for determining whether the flare is received or rejected by the seeker. The flare travels and is observed for a time SKFLRJCTTIME, during which it is affecting the seeker. After that time, a random draw against a number SKFLRJCTPROB ascertains if the flare is accepted or rejected. The time and probability are stored in SMLIB for each IR missile type. A better scheme would be to use either the kinematics of the flare or the spectral properties of its radiation for the rejection process. Since we do not have data on either of these, we have resorted to the overly simplified process just described. The numbers for time and rejection probability at present in SMLIB are quite arbitrary, and any conclusions based on them should be taken cum grano salis.

If the flare is rejected by the seeker, it continues to fly, but its radiation does not affect the seeker. If it is accepted, the seeker now has as inputs the radiation from the target and the flare. The weighted sum of the angles to each sets the pointing angle of the seeker toward the center of radiation. If either the target or the flare moves outside the field of view of the seeker, its radiation is set to zero.

Most often, a flare is brighter than an aircraft, and the seeker is decoyed.

The aircraft continues to release flares, following its specified pattern, at the time separation IRFLINT (infrared flare interval). Each flies until it burns out at the end of its flight interval, set by the length of the lookup tables for drag and brightness. For the flare type used in RJARS, this duration is six seconds. Each flare undergoes the rejection test. The seeker will point toward the resultant center of radiation of the signals from the aircraft and all the accepted flares, each at its proper flight time. An index ASNFLARE marks the number of flares from each aircraft. A subroutine KILLFLARE tests if the flare has reached the end of its burn time or has hit the ground. If it does either, the index of any other flare from that aircraft is decremented (the nature of the dispensing ensures that the burnout of flares is sequential with index). The SAM flight continues until the endgame is reached.

5. Endgame

SAM flight data are printed out under the options SMPRNT and IPG. If SMPRNT = 1, then detailed flight data appears at each IPG'th subdivided time step. Otherwise the SAM state and range to its launcher and target are printed each time step. When the range to the target is less than 1000 feet, the detailed data are printed every subdivided time step. Printed quantities are listed in the user's section.

The SAM continues its flight until any of several conditions is met:

1. Its range from the launcher exceeds the range of the aircraft from the launcher. This is usually the final condition when the following tests are not met.
2. The range from the SAM to its target or the partner of its target is less than the fuzing range SMFUZRNG and the range to the closer of the indicated objects is increasing. The fuze range is stored in SMLIB.

3. The SAM uses IR guidance and the seeker has lost lock on its target. The target may change during the flight.

When any of these tests is passed, the endgame flag SMENDFLG is set, and the position of the SAM is interpolated backwards along the last segment of its flight to determine the time and the coordinates of the point of closest approach to the true position of the aircraft. The flight of the SAM has used imperfect radar data, but to this point has been treated as following its guidance system perfectly. Guidance errors are now simulated by adding a random vector to the miss distance vector. The magnitude of this vector is normally distributed with a standard deviation equal to the input value SMCEP, and the direction of the vector is uniformly distributed over the unit sphere. If the SAM is a shell, the SMCEP stored in SMLIB is actually an angular dispersion, and it is multiplied by the range to give the displacement SMCEP. In either case, the resulting magnitude of the miss distance is then calculated.

The aircraft is approximated as an oriented ellipsoidal fuselage with flat elliptical wings. The fuselage is circularly symmetric around the body longitudinal axis, with an elliptical side view with major axis ACLENGTH and minor axis ACFUSDIAM. The elliptical wings have major axis ACSPAN and minor axis ACWINGCHORD. All these dimensions are stored in ACLIB. If the location of the miss distance calculated above is within the fuselage or wing, a direct hit, the kill probability is set to unity. When a suitable model of kill probability versus hit probability becomes available it will be incorporated in RJARS.

SAMs with contact fuzes can kill only with a direct hit. For those SAMs with proximity fuzes, the endgame kill probability in RJARS is assumed to be a function of the magnitude of the miss distance from the nearest critical point on the oriented aircraft. Warheads generally produce both blast and fragmentation effects, so the kill depends on the orientation of the vulnerable components of the aircraft. An attempt to simulate these effects leads to great complications (see Ref. 6, pp. 158-204). At present, the kill probability versus distance is taken to depend on the SAM type but not on the aircraft type, and the angular dependence of the kill probability has been dropped.

The kill probability is the product of the probability that the warhead detonates (library input) and the probability a kill is achieved if it does detonate. At close distances, below a range KRMIN, blast effects are so strong that the latter probability is unity. At great distances, beyond a range KRMAX, the kill probability is zero. We have approximated the kill probability as a linear function of range between KRMIN and KRMAX (both library inputs). This type of relation is consistent with test results on various SAMs.

The calculated kill probability is compared to a random draw to determine the outcome of the endgame. The success or failure is printed and stored for the summary. If the kill is successful, the aircraft's flag ASDEAD is set to zero, which removes it from the simulation. Otherwise the SAM enters the F or refire state for another shot. The number of missiles available at the state is checked as described at the beginning of this section.

If a radar-guided SAM has been deflected to a decoy and kills it, the aircraft will release another decoy, as long as there are still decoys on board. The launcher will think it has missed the target, and will begin to count down another SAM. The decoy identification number is incremented to make clear to the summary just what has happened. If an IR-guided SAM is deflected to a flare, it will kill it, leaving the aircraft unscathed. Flares still in the air will continue to fly until they burn out or crash. If available, a new SAM will begin countdown, and it is assumed to be unaffected by flares until it launches. The possibility that the optical observer associated with the SAM can identify flares has not been implemented.

I. UPDWR--UPDATE WARNING RECEIVERS

The warning receivers in RJARS are used to establish whether the aircraft is detecting the radar and whether the jammer on the aircraft should be turned on or off. The receiver contains WRNBAND frequency bands, each of which has an upper and lower frequency limit and a receiver sensitivity WRBSENS. Most airborne warning receivers use multiple antennas to achieve coverage without scanning. Each antenna provides coverage in a particular direction. The signals from the several antennas are combined, and the resultant displays gain without scanning. The combined pattern is approximated by a "rosette" antenna pattern, which contains WRNLEAF lobes. All lobes have the beamwidths WRNHB and WRNVB in the horizontal and "vertical" planes and are offset by an angle WRDIP in the vertical plane. The first lobe is offset from the forward direction by an angle WRBS, and the succeeding lobes are centered on directions displaced from the first by multiples of an angle WRPHIN = 360/WRNLEAF.

1. Power Calculations

A flag WROUTBFLG determines if a radar lies within the frequency coverage of the warning receiver. If this flag is set, then that receiver ignores that radar thereafter. Otherwise, at each time step the receiver on each aircraft checks the signal from each radar that is transmitting, is within the horizon, and is within the maximum range of the radar against that aircraft. The last choice is to ensure that the jammer will not be turned on if the radar cannot detect the aircraft. The power from the radar is calculated from the equation:

$$P = \frac{(ERP)_R * G_W * C^2 * (SDL)_{RW}}{(4\pi)^2 * F_R^2 * R_{RW}^2} \tag{42}$$

where G_W is the gain of the warning receiver antenna in the direction of the radar, $(SDL)_R$ the sidelobe level of the radar in the direction of the receiver, R_{RW} the range from the radar to the receiver, and the other symbols are familiar. For search or acquisition radars, the sidelobe level is set equal to unity, corresponding to the time at which the radar scan sweeps over the receiver. For a tracker pointing at the selected aircraft, the sidelobe level is again unity, but for a tracker pointing at some other aircraft the appropriate sidelobe level must be used. The gain of the receiver antenna is calculated from the rosette pattern.

The received power from Eq. (42) is compared to the receiver sensitivity in the proper frequency band. Receiver sensitivities are customarily expressed in decibels with respect to one milliwatt (dBm), and RJARS adheres to this conversion. In reality, many radar pulse trains may simultaneously be incident on the receiver, and the separation of these trains and identification of signals is a very difficult task. RJARS ignores these "high density environment" problems, and assumes that the several signals are immediately identified.

This is a possibly significant limitation in the model. One resolution might be to provide a delay in identification that is proportional to the number of signals received. Another is to allow for saturation of the receiver, by limiting the number of signals it can accept in any frequency band. Still another is to limit the number of pulses per second that can be handled, thereby limiting the number of radars, with special provision for high repetition rate pulse doppler transmitters. These are all possibilities for future expansion of RJARS.

2. Receiver Decisions

After the comparison with the sensitivities there are a number of possibilities. A flag WRRDFLG marks whether the receiver has detected the radar. WRRDFLG has as arguments the aircraft identification number and the radar identification number, so there is a flag for each aircraft-radar pair. For simplicity the variable will be used without citing the arguments. If WRRDFLG = 0 and the received power is below sensitivity, then the undetected radar is still undetected, and nothing need be done. Also, if WRRDFLG = 1 and the received power exceeds sensitivity, then the catalogued radar is still being detected, and again no procedure is necessary.

If WRRDFLG = 0 and the received power exceeds sensitivity, then the receiver has detected a new radar. It adds one to the running count of radars detected in that frequency band, sets the flag WRRDFLG to 1, and sets WRBID, the array item that tells the receiver which radar it is dealing with, to the radar identification number. It then checks if this radar type is one that it should jam. If true, then if the radar is a long-range searcher, or if it is an acquisition or track radar either pointed at the aircraft itself for aircraft that are not members of a group or pointed at itself or a member of the aircraft's associated group, then the subroutine WRJMON is called to begin jamming the radar.

If the flag WRRDFLG = 1, and the received power is below sensitivity, then there are further possibilities. If the radar is a search or acquisition radar, then the parameter INBEAM is checked to ascertain if the radar has swept over the receiver during the last time step. If it has not, then the situation is only temporary, and the receiver does nothing. If it has or if the radar is a tracker, then either the radar went off the air or the aircraft went over the horizon or beyond detection range. If the radar is off the air, the receiver and jammer should be turned off, a task performed by the subroutine WRRECOFF. The program calculates receiver power only for those radars which are not over the horizon. If the received power has dropped below sensitivity while the jammer is on, as could happen if the radar is a tracker aimed at another member of the aircraft's group, so the aircraft is in the sidelobes of the radar transmitting antenna, then also the receiver and jammer should be turned off. If the radar and aircraft are mutually masked by terrain, then the jammer should be turned off regardless of the received signal.

3. Jammer Decisions

The subroutine WRJMON is called when the receiver is turned on or when the aircraft joins a group that is jamming the radar. WRJMON first checks that the radar is within the frequency coverage of the jamming transmitter, and if so selects the proper band. It then determines what is the most effective jamming technique to use against that radar, and looks at its own stable of techniques to find if the desired technique is available. If so, that technique is the jamming tactic to be employed; otherwise barrage noise is radiated. The jammer flag JMRDFLG is set, signifying that particular radar is being jammed, the count of radars in the band being jammed is incremented, and the identification JMBID is set to the radar number.

The jammer antenna is approximated by a rosette, just like the receiver. The parameters of the rosette will be different. For those jammers that use fixed antennas, the actual combined rosette pattern will be used. For those jammers that scan the antenna and track the target, the number of leaves of the antenna is set to one. In such cases, the power radiated by the jammer is emitted as if the jammer antenna is pointed directly at the radar, corresponding to accurate measurement of the angle of arrival of the signal followed by steering of the jammer transmitter antenna. If the jammer antenna gain is low, then the situation could also correspond to a broad antenna beam which encompasses the radars being jammed or even an isotropic transmitter. Note that the ERP of the jammer should be interpreted as the product of the on-axis antenna gain and the power input to the antenna, including line and reflection losses. RJARS divides the power available in a given frequency bond equally among all the radars in that band that are being jammed at a given time. A priority designation would be desirable but has not been implemented.

The subroutine WRRECOFF turns the receiver and jammer off when called. It first identifies which position in the count of signals in that band is occupied by that radar. The positions are determined by the sequence in which the radars were detected. Those radars that occupy higher positions are each lowered one position and the now vacant highest position is dropped. The count is decremented, and the receiver flag is turned off. The subroutine WRJMOFF is then called to apply the same procedures to the jammer. The event information is printed.

UPDWR is the last calculation routine called during the simulation run. After its completion, the time is incremented and the procedures repeated until either the final time is reached or there are no surviving aircraft (the possibility of no surviving radars is very unlikely. If it were to occur, the aircraft would simply continue flying until the time is up.) At the simulation end, the subroutine output is called to print the summary of the run.

J. OUTPUT--PRINT SUMMARIES

1. Print Sequence

OUTPUT first identifies which run in the Monte Carlo sequence is being printed. It then prints the search radar summary, specifying the behavior of each search or acquisition radar against each aircraft. The number of looks of the radar at the aircraft, number of hits (successful detections), and their ratio is printed, giving a measure of radar and also jammer effectiveness. Each time and range of entry or exit of the aircraft into the field of view of the radar is printed (there may be many if the aircraft is frequently masked by terrain), and the time and range of each detection and loss of detection. Finally, the average number of radars pointed at each aircraft, as determined by the number of times radar beams passed over the aircraft divided by the number of time steps is presented.

The second summary pertains to trackers. For each tracker aircraft combination which actually occurred (the parameter RTITK counts the number of times a tracker was assigned to an aircraft), the times of assignment, turning-on and turning-off of the acquisition radar, and turning-on and turning-off of the tracker are printed. The anomalies described in the earlier version of RJARS have been corrected. All times are properly initialized and updated each time step, so all printouts are consistent. It is still possible for the acquisition radar to turn the tracker off without its ever having been on. If this occurs, the tracker on and off times are the same as the acquisition off time.

After the trackers, the summary of the SAMs is printed. For each SAM which reaches the launch state, the identification of the launcher, the list number for that SAM among the several SAMS fired from that launcher, the target aircraft, the launch time, and the launch outcome (success or failure) is printed. If the SAM is decoyed, the identification number of the decoy is printed. If the SAM crashes or loses tracking during flight, that result and its occurrence time is printed. If the SAM performs an interception, the time, range from the launcher, miss distance, kill probability, and outcome (MISSED TGT or SHOT TGT DOWN) are printed. Following is a box score showing the number of SAMs launched, launch failures, number losing track, number of misses, and number of kills.

Next comes a summary on the anti-radiation missiles. For each ARM or each aircraft its identification and whether it has been launched (YES, NO) is printed. For those that have been launched, the launch time, target radar, and flight result (LOST TRACK, REACHED TGT) are printed. If the ARM reached the target, the kill time, kill probability, and outcome (MISS, KILL) are printed. If there are no ARMs on board, this summary is omitted. If there are no ARMs on any aircraft, the header is omitted.

The following aircraft summary shows for each aircraft if it reached its destination, if it was shot down on ingress or egress, and the time of each event. The surviving aircraft may still be in the field or may have escaped safely. The appropriate expression is printed. For each aircraft, its number of engagements with SAMs is printed. A box score shows the number of aircraft that entered the field, number shot down on ingress, number reaching destination (a measure of mission effectiveness), number shot down on egress, number still in field, number escaped, total number of engagements with SAMs, and the average number of SAM engagements per aircraft.

2. Statistical Calculations

If the RJARS simulation is in cooperation with a JANUS operation, statistics on the aircraft survivability are collected in OUTPUT and prepared for transmission to JANUS. Even if JANUS is not used, these statistics may be of interest to the user. The decision to collect these statistics is controlled by the flag IJANUS (1 collect, 0 do not collect). The statistics are contained in the structure PRDBN.

The concept behind the probability distribution is that the aircraft interact with the ground defenses in a sequential manner. An aircraft enters the operating region of a defense, flies through it, and is either killed or escapes. There may be several shots at the aircraft from a given launcher, determined by launch success and kill probability. Thus, there is a time interval for the aircraft and defense to interact, and a smaller time interval within which a kill may be achieved. For each interaction on each iteration, the kill time, if any, is stored. At the end of the simulation (JREP = NREP), the earliest and latest kill times for each interaction determine the start and end times of the killing interval. To a reasonable approximation, the kill probability density is uniform over this interval. The set of intervals, together with the kill probability per interval, is calculated for each interaction and stored for transmission to JANUS. If the defensive system is dense, several elements may be shooting at the aircraft at the same time. The corresponding kill intervals will overlap, and the resultant kill probability densities are additive for the overlapping times. Also, there may be more than one interaction interval between an aircraft and a defense. All such intervals are recorded separately.

The data transmitted from JANUS via CAGIS to RJARS contain two times, designated LOS1 and LOSN. The first is the time at which an aircraft can first perform its mission, the second the time at which it leaves its target area. The times are located in the flight planner as initial data. The JANUS operators were interested in the kill probability before LOS1, which gives the likelihood that an aircraft was killed on ingress, the detailed intervals between LOS1 and LOSN, and the kill probability after LOSN, the egress kill probability. All these quantities are calculated and transmitted.

When the OUTPUT subroutine has printed the summaries, the Monte Carlo subroutine UPDMC is called to determine if the run should be repeated. If the last run has been completed, the program prints "RJARS is finished" and stops. Otherwise, the program flags and variables are returned to their initial values, except for the random number generator, whose present value is printed at the beginning of the next run. RJARS then executes another simulation.

K. CALCSR--AUXILIARY-CALCULATIONAL SUBROUTINES

RJARS contains seven mathematical subroutines, UNIV, NORMV, RECT, SPHER, LINSGHT, TRINDEX, and ZTERN. The subroutine UNIV is the random number generator, which produces a variable YFL that is uniformly distributed between 0 and 1. UNIV operates on the multiplicative congruence principle. A number XR is multiplied by a fixed multiplier A1. The product is divided by a number M1 and the remainder is taken as the next value of XR. XR begins with an input value SEED, and is permitted to take on only odd integer values. The product XR * A1 will usually exceed the value of M1, and the remainder after division should be uncorrelated with the previous value of XR. Thus, the values of XR should be uniformly distributed over the range 1-M1, with no correlation between successive values. Dividing XR by M1 produces YFL, uniformly distributed between 0 and 1. In RJARS the value of M1 is 2147483647 = $2^{31} - 1$, the largest number that can be carried on the IBM-3033 operating at RAND, and A1 = 62089911, a number that produces a sequence of maximum length and minimum correlation between successive values. This choice of parameters should produce a sequence approximately 2^{28} in length before a repetition is likely (M1/4 is the theoretical value, half that because of the limitation to odd values). This should be long enough fo practical purposes.

The subroutine NORMV uses UNIV to calculate two independent, normally distributed variables of mean zero and variance 1. It is derived as follows. Let X and Y be the two variables. Then their probability density is

$$P(X,Y) = \exp(-(X^2 + Y^2)/2)/2\pi \tag{43}$$

with R and θ the polar coordinates corresponding to X and Y, their probability density is

$$P(R,\Theta) = R*\exp(-R^2/2)/2\pi \tag{44}$$

thus the phase is a random variable uniformly distributed over the range 0-2π, and the amplitude has a Rayleigh distribution. The cumulative distribution of R, that is, the probability that R exceeds some value A, is:

$$W(A) = 1 - \exp(-A^2/2) \tag{45}$$

The variable $\exp(-A^2/2)$ thus is uniformly distributed over the range 0-1, since that is the character of a cumulative distribution. The normally distributed variables are calculated by inverting the procedure that led from Eq. (43) to Eq. (45). Thus, call UNIV, then set VAR1 (a calculation variable equivalent to A) to the value:

$$\mathrm{VAR\ 1} = \mathrm{sqrt}(-4.60517*\mathrm{LOG\ 10(YFL)}) \tag{46}$$

(Since LOG10 is the only logarithmic base used elsewhere in RJARS, we did not add the natural logarithm to the built-in mathematical functions list. This is a residue of the PL-I origin for RJARS.) Then call UNIV again, and set the variables VAR2 and VAR3, corresponding to X and Y, to

$$VAR\ 2 = VAR\ 1 * \cos(2*PI*YFL) \tag{47a}$$

$$VAR\ 3 = VAR\ 1 * \sin(2*PI*YFL) \tag{47b}$$

This pair of independent normally distributed variables is used in the numerous calculations of errors.

The subroutine RECT, which has been copied from JARSM with minor correction, converts from earth-centered latitude-longitude coordinates to rectangular coordinates in the plane tangent to the earth's surface at the center of the coordinate system. The inputs to RECT are latitude and longitude in degrees, minutes, seconds, and hemisphere, the outputs are the rectangular coordinates in nautical miles. The derivation is in App. B of Ref. 2. If P is the point in question, R the reference point, then the rectangular coordinates X,Y are given in terms of the latitude and longitude of P and R by the equations:

$$X = 3440 * \cos(\text{Lat } P) * \sin(\text{Long } P - \text{Long } R) \tag{48a}$$

$$\begin{aligned} Y = 3440 * [&\sin(\text{Lat } P) * \cos(\text{Lat } R) \\ &- \cos(\text{Lat } P) * \sin(\text{Lat } R) * \text{abs}(\cos(\text{Long } P - \text{Long } R))] \end{aligned} \tag{48b}$$

The coefficient 3440 is the earth's radius in nautical miles. All calculations in JARSM and RJARS are performed in the rectangular coordinate system. When the inputs are in the global latitude-longitude system, RECT and Eqs. (48a) and (48b) are used to convert the input aircraft and radar positions to rectangular coordinates.

RJARS can handle both input and output in both coordinate systems. The choice is established by the parameter COORD, also used in JARSM, which here takes on the four values

COORD = 1	Input and output rectangular
COORD = 2	Input and output global
COORD = 3	Input rectangular, output global

COORD = 4 Input global, output rectangular

To convert from rectangular to global, a subroutine SPHER is used, which inverts Eqs. (48a) and (48b). The process leads to a quadratic equation whose roots are:

$$\left[Y^* \cos(\mathrm{Lat}\ R) \pm \sin(\mathrm{Lat}\ R)^*\mathrm{sqrt}((3440)^2 - X^2 - Y^2)\right]/3440 \tag{49}$$

The inverse sine of these values is calculated, and the one that is closer to Lat R is the latitude of the point P. The longitude is then given by:

$$\mathrm{Long}\ P = \mathrm{Long}\ R + \mathrm{asin}(X/(3440^* \cos(\mathrm{Lat}\ P))) \tag{50}$$

The latitude and longitude as outputs are expressed as degrees and decimal fractions of degrees.

Finally, the calculational subroutine LINSGHT calculates the range, azimuth, and elevation of an aircraft from a radar. It takes the coordinate differences DX, DY, DZ, and converts these to spherical coordinates centered at the radar. If terrain is not used, or if the aircraft is indicated as visible to the radar by the terrain model (if the terrain details are derived from maps, the aircraft may be indicated as visible when it is actually below the curve of the earth), LINSGHT determines if the aircraft is over the horizon and sets the flag RDACOH.

TRINDEX identifies the indices of an element in the terrain file TERRA when its location is given. Each coordinate of the element, measured from the origin of the rectangular coordinate system or LATO, LONGO for the global coordinate system, is divided by the terrain element size in that coordinate direction. The terrain elements are centered on integral values of the indices, and extend half an element length in each direction. The magnitude of the quotient is compared with the greatest integer contained in it. If the remainder after comparison is less than .5, the integer equals the index magnitude. If the remainder exceeds .5, the next greater integer is used. The value of the integer after this decision is multiplied by the sign of the coordinate to give the correct index value.

The subroutine ZTERN calculates the terrain height at any position in the field. It first uses TRINDEX to find the indices of the terrain element in which the point lies. The normalized displacements x and y from the center of the element are found by subtracting the coordinates of the center from the coordinates of the point, then dividing by the appropriate element length. The height of the terrain surface is fitted by a quadratic expression that uses the element and its eight neighboring elements. Let i and j be the indices belonging to the point in question. Let z be the height at any index point, a function of the indices, and h be the height at the field point. Then h is given by the equation:

$$\begin{aligned} H = z\,i,j) &+ (z(i+1,j) - z(i-1,j))*x/2 \\ &+ (z(i,j+1) - z(i,j-1))*y/2 \\ &+ (z(i+1,j) - 2*z(i,j) + z(i-1,j))*x^2/2 \\ &+ (z(i,j+1) - 2*z(i,j) + z(i,j-1))*y^2/2 \\ &+ (z(i+1,j+1) - z(i+1,j-1) - z(i-1,j+1) + z(i-1,j-1))*xy/4 \end{aligned} \tag{51}$$

In this expression, the first term is the contribution from the center, the second and third represent the terrain slopes in the two directions, the fourth and fifth represent the terrain curvature in the two directions, and the last corresponds to the orientation of the principal axis of the ellipsoid that has been fitted to the terrain. Alternatively, the second through the fifth represent the nearest axial neighbors of the element, and the last the asymmetrical interaction of the nearest diagonal neighbors. If a point within a border element of the terrain is to be treated, this equation would overrun the terrain array, which is why no aircraft or radar can be permitted to be within a border element. At the beginning of the simulation, RJARS builds a rim around the apparent terrain border to avoid this problem.

Besides these mathematical subroutines, CALCSR contains a number of the calculational subroutines referred to in other sections. Descriptions of these subroutines follow.

The subroutine ORIENT takes two directions in inertial space, and finds the angles between the second and the first. Thus, it finds the offset angle of a seeker from its boresight angle. The mated subroutine REORIENT combines the angle from the second to the first with the inertial coordinates of the first to find the direction of the second vector in inertial space.

The subroutines SEKRPAT and MONOPAT find the patterns of a monopulse antenna as described previously.

The subroutine FACE starts with two points on the ground. It finds which terrain elements are intersected by the line connecting the points. The number of such elements and the length of the line in each element are found. For each such element, the terrain indices, the coordinates of the midpoint of the connecting line, and the type of terrain are indicated.

The subroutine CLUTTER uses FACE to find the relevant terrain elements. For each, the azimuth and elevation to the radar are found, self-shadowing is determined, and the cross-section per unit area is calculated from the fundamental terrain properties, the radar frequency, and the angle of incidence.

The subroutine RESPNS finds the gain for a monopulse antenna in the direction of the clutter point when it is aimed at the target. The subroutine POWER finds the total return from a clutter element into a monopulse seeker, using the geometry and RESPNS.

The subroutine ISORAD locates the ground points that satisfy the conditions for pulse doppler clutter, and finds the effective area of the clutter regions.

The subroutine MASK ascertains if two points are mutually visible. The subroutine RIDGES calculates the locations of the successive ground entries to and exits from visibility, providing the range and depression angle of each such point and the total number thereof. The subroutine RIDGTEST takes a given range and azimuth and ascertains if the ground is visible to the radar at that range and azimuth.

The subroutine MULTIP calculates the multipath effects, using many local variables. First FACE is used to find the terrain elements between the target and the radar. Self-shadowing is checked, and RIDGTEST determines if the element is visible to the radar. MASK finds if the aircraft can see the element. If both can see, the slope of the ground at the point along the connecting direction is calculated from the terrain data. The angles of incidence of both rays with the element are found and tested for positive grazing angles with respect to the local slope. The directions at the edges of the terrain element are found and the location where the difference between the angles of incidence passes through zero is ascertained, if such element exists. The specular point will be in that element. The reflection coefficient at that point is found in terms of the terrain parameters, the roughness, and the angle of incidence, using Eq. (21). For each specular point, the path length in the element and the phase of the ray reflected by multipath relative to the direct ray are found, yielding an effective complex reflection coefficient. The gain of the antenna toward each element is found, and the net signal (real and imaginary components multiplied by normally distributed random variables to take care of the independence of the reflections) is found for each element and summed to give the total multipath signal.

This completes the description of the analytical model used in RJARS. The next section is a user's manual describing how to prepare files and operate the program.

IV. RJARS USER'S GUIDE

RJARS is a non-interactive program that operates entirely from input files and delivers its output to the standard printout of the computer system and to the file for graphical display. Consequently, the initial task of the user is to prepare the input files. As discussed, these are of two types; library files that contain equipment data, and simulation files that provide the values of the variables associated with a particular run. Most of this user's guide is devoted to a description of how the files are prepared.

There have been many changes in the files since the earlier version of this report [Ref. 3]. The description parallels the previous work, but it should be emphasized that only the current files can be used with the operating version of RJARS.

PREPARATION OF LIBRARY FILES

We shall discuss the nine library files first in alphabetical order. They can be prepared at any time, and updated as information on additional equipments becomes available. Library files cannot be updated during a run. All library files begin with descriptive header lines. Except where specifically stated, all entries are separated by commas, including a comma at the end of each line.

The library files are:

1.	ACLIB	Aircraft performance data
2.	ACRCS	Aircraft cross-section data
3.	ARLIB	Anti-radiation missile data
4.	IRLIB	Infrared and optical data
5,	JMLIB	Jammer data
6.	MCLUT	Terrain properties
7.	RDLIB	Radar data
8.	SMLIB	Surface-to-air missile data
9.	WRLIB	Warning receiver data

1. Aircraft Parameters

The first file, ACLIB, is new. It contains the data that represent the physical properties of the aircraft, mostly dimensions and acceleration capabilities, plus some special radar cross-section data for helicopters. All the data are placed on a single line for each aircraft type. An initial header line identifies the variables in sequence. In the order of their appearance on the line, the variables contained in ACLIB are:

ACTYP	Aircraft type (integer)
ACNAME	A character string naming the aircraft. This variable is not used in RJARS, but serves to relate the aircraft type to real aircraft. The string must contain no blanks and be followed by a space.
ACTURNAC	Maximum acceleration in the yaw direction during a turn (g's)
ACPITACUP	Maximum acceleration in the upward pitch direction (g's)
ACPITACDN	Maximum acceleration in the downward pitch direction (g's)
ACLENGTH	Length of the fuselage (feet)
ACSPAN	Span of the wings, tip to tip (feet)
ACFUSDIAM	Diameter of the fuselage (feet)
ACCHORD	Chord of the wing at its root (feet)
ACTAILAREA	Area of the vertical tail (sq feet)
ACROTORRCS	Radar cross section of the helicopter rotor, edge on, horizontal plane (dBsm)
ACSENSORRCS	Radar cross section of a helicopter externally mounted sensor (dBsm)
ACSKIDFUSE	Height of the center of the helicopter fuselage above the skid (feet)

2. Radar Cross Section

The second file, ACRCS, contains the radar cross section (rcs) data for each aircraft type. The format has been modified from the earlier radar cross section file to include three dimensional data and variable resolution. The first data line for each aircraft type includes four items:

ACTYP	Aircraft type
ACNAME	See ACLIB
ACRCSAZRES	Resolution in azimuth of the rcs data (deg)
ACRCSELRES	Resolution in elevation of the rcs data (deg)

The azimuth and elevation resolutions are independent and can differ from aircraft to aircraft, but for any given aircraft type, all the data corresponding to that resolution must be present. Thus if ACRCSAZRES equals 10 degrees, the rcs must appear for each elevation value at all of the azimuths 0, 10, 20, ... , 170, 180. The data are arranged so a given line corresponds to a fixed azimuth, beginning with zero at the nose and ending with 180 at the tail. The values along that line give the radar cross section as a function of elevation at that azimuth. The units of radar cross section are decibels relative to 1-m^2 (dBsm). The elevation steps in units of the elevation resolution, beginning with zero elevation corresponding to observation from below, and ending with 180 for observation from directly above. The azimuth and elevation resolutions must each divide into 180, so an integral number of elements appear on a line. For example, if the azimuth resolution is 5 degrees, and the elevation resolution is 10 degrees, there will be 37 lines of data (180/5 + 1), with 19 entries per line. If the data from which the file is derived do contain a full range of elevation, there is no problem with data entry. If, as often occurs, there exist only data in the horizontal plane of observation, then that data should appear in the central vertical column of the rcs file, and zeroes should appear in the rest of the table. In such a situation the variable NDOF in the scenario should be set to 3 to avoid misinterpretations (RJARS will think the aircraft has a 1 sq meter cross section in directions other than the horizontal plane).

The variable in the data entries of ACRCS is the RJARS variable

ACALPHA1	Radar cross section, azimuth data only (dBsm)
ACALPHA2	Radar cross section, azimuth and elevation data (dBsm)

3. Anti-radiation Missiles

The third file, ARLIB, holds the data on anti-radiation missiles. It begins with two header lines to describe the variables. Then for each missile type we first list a line that contains the name or description of the missile, then a line with the following variables:

ARLTYP	Number identifying missile type
ARLNBD	Number of frequency bands in receiver
ARLVEL	Velocity (fps)
ARLRMIN	Minimum operating range (nmi)
ARLRMAX	Maximum operating range (nmi)
ARLTDLY	Time to fly without guidance (sec)
ARLLOFT	Initial loft angle (deg)

Next come lines for each frequency band of the receiver. On each line are the variables

ARLBDNO	Number of the frequency band in question
ARLBDLFR	Band lower frequency limit (MHz)
ARLBDUFR	Band upper frequency limit (MHz)

After all ARLNBD lines have been entered, the next missile is entered. Only five ARM types should be included in ARLIB. This is not likely to be a serious restriction.

4. Infrared Parameters

The next file is the new file IRLIB. It contains all the data that pertain to infrared and optical equipment. It begins with six header lines that describe the data in the file. The first data line contains the ranges IRRANGE at which attenuation data is stored. There are six values on this line, given in nautical miles and corresponding to 0, 1000, 5000, 13000, 30000, and 100000 feet. The next two lines pertain to the solar radiation and attenuation. For each, the first entry is the solar amplitude in watts/ster in the appropriate wavelength band IRSUNSTR, and the next six are the attenuation of the solar radiation IRSUNATTN at the ranges indicated above. Note that the attenuation in IRLIB is a multiplicative factor on the amplitude, not a coefficient in an exponent.

Next is data on paint. A header line indicates the variables. Each line contains three variables--an index counting the types of paint, the variable PAINT, a two-character code, and the reflectivity of that paint. The last entry in the paint list must have index zero and PAINT type NR (nonreflecting).

After another header comes a single line containing the values for the matrices SKYGND and EXTINCT that describe the attenuation of optical contrast with range. There are four values for each, corresponding to the two indices ISUMMER and ISEEING.

The extensive flare data follow. First are two description lines, the first showing that flare data is to appear, the next identifying the three variables that follow on the succeeding line:

IRFLREL	Flare launch reliability
IRFLVEL	Flare launch velocity (fps)
IRFLINTVAL	Interval between launches from a dispenser (sec)

Next are flare brightness values. After a descriptive line, a line gives the 15 values of the variable IRBRITETIME, the times at which the brightness of the flare is specified. Another line contains the three values of the variable IRBRVEL, the velocities at which the brightness is specified. Then come 15 lines with three values each, the values of the flare brightness IRBRITE as a function of time and velocity. After a descriptive line specifying the bands are two lines with the values of the attenuation IRFLATTN of the flare signal in each band at the usual specified ranges.

The flare mass and product of area and drag coefficient are given as functions of time and velocity in the next four lines. After a descriptive line comes a line with the four values of IRCDTIME and the two values of IRCDVEL (knots). On the next line is the mass IRFLMASS (kg) at the four times, and on the following line is the drag product IRFLCDS in the sequence: two velocities at the first time, two velocities at the second time, etc. This completes the flare data.

Next is a set of attenuation tables. As discussed in the analysis section, these tables and the following signal tables will be changed to incorporate a better propagation model. After a descriptive line showing that attenuation tables are coming and a second describing the first spectral band is a set of five tables for five ranges, with a descriptive line indicating the range preceding each table. The tables give the values of the attenuation IRATNFAC as a function of elevation and azimuth at that range in that spectral band. The data on a line are at a specified elevation, with the first corresponding to observation from below and the last to observation from above, in steps of 15 degrees. The data on that line are the values of IRATNFAC in azimuth steps of 15 degrees from tail to beam. It is assumed in the program that the attenuation in azimuthal directions between the beam and the nose is the same as at the beam, so only half the full azimuth range need be covered. Note the change in sequencing between radar data and IR data, a consequence of historical precedent. After the full set of data is listed for the first spectral band, there is another descriptive line for the second spectral band and a corresponding set of attenuation data.

As discussed in the analysis section, the same attenuation data are used for all aircraft, a situation that will change in the future. The next set of entries in IRLIB, following a single descriptive line, pertains to the IR data for the various aircraft types. For each aircraft type, the first line contains a set of miscellaneous variables:

ACTYP	Aircraft type
ACNAME	Descriptive string showing what is to follow. See ACLIB for conditions.
ACNOSEDET	Probability of detecting an IR target directly ahead
ACTAILDET	Probability of detecting an IR target directly behind
ACFLTILT	Tilt of flare launcher from vertical (deg)
ACFLAZ	Azimuth of each flare launcher from forward (deg)
ACFLFUSDISP	Longitudinal displacement of flare launchers from center of nose-to-tail line (ft)
ACFLWINGDISP	Lateral displacement of each flare launcher from center line (ft)
ACLFLARE	Number of flares per dispenser

Two tables that follow give the aircraft radiated power in watts per steradian in each spectral band as functions of elevation and azimuth. After a header describing the band, the sequencing follows the attenuation tables, except that the full range of azimuth is covered. The entry item is the variable ACIRSIG.

The aircraft data as described complete the IRLIB file. If an aircraft is called in the scenario and there are no IR data available for it, RJARS will halt with an error indication. It may be necessary to improvise data. The accuracy of such improvised IR signal data cannot be confirmed by RJARS. Remember, if new values are to be inserted, the numbers called for are watts per steradian, not temperature. It is only fair to note that much of the data in the working IRLIB file used at RAND have been found by interpolation from some very sketchy measured values.

5. Jammers

The next file, JMLIB, contains the jamming transmitter data. It contains a single header line to identify variables. Then for each jammer type the first line contains the information:

JLTYP	Number identifying jammer type
JLCODE	5-character code describing jammer
JLNBAND	Number of frequency bands in jammer
JLRDTECH(1,2,3,4)	Array of 4 values, each equal to 0, 1, 2, 3, or 4, listing jamming techniques available to the jammer. If the value 1 appears in the array, the jammer can perform search deception; 2, tracking range deception; 3, tracking angle deception; 4, tracking range and angle deception. The variable JLRDTECH(0) is set internally to 0 so all jammers can radiate noise.

Data for each of the NJLBAND bands of the jammer follow. NJLBAND cannot exceed 20. For each band line the variables are:

JLBNUM	Number identifying band
JLBCODE	3-character code describing band, e.g., low
JLBERP	Effective radiated power (W)
JLBLFRQ	Lower frequency limit of band (MHz)
JLBUFRQ	Upper frequency limit of band (MHz)
JLBBDW	Bandwidth of radiated noise (MHz)
JLBPOL	Polarization loss (dB, positive value = loss)

A blank line separates the last band line of one transmitter from the first line of the next. This description pertains to JMLIB before the inclusion of repeater decoys. A file modification adds to the end of the first line the variable JLGAIN, the gain of the jammer if it were used as a repeater. For most jammers, this variable is not used and is set to unity. Also, a recent modification adds at the end of the data line for each band the variable JLBREL, the reliability of that band.

6. Terrain Parameters

Next consider the new file MCLUT. This contains the properties of the ground used for clutter, multipath, and optical reflectivity. After a descriptive header, the data for each terrain type are listed on its single line. For the interpretation of the variables see the analysis. The sequence of variables is as follows:

TRPTYPE	Index, terrain type
TRPCODE	Terrain type description (5-character code)
TRPA	Parameter A in terrain radar cross-section
TRPB	Parameter B in terrain radar cross-section
TRPC	Parameter C in terrain radar cross-section
TRPD	Parameter D in terrain radar cross-section
TRPROUGH	RMS roughness parameter (cm)
TRPEPSR(0)	Real part of dielectric coefficient (dry)
TRPEPSI(0)	Imaginary part of dielectric coeff (dry)
TRPEPSR(1)	Real part of dielectric coefficient (wet)
TRPEPSI(1)	Imaginary part of dielectric coeff (wet)
TRPREFL	Visible reflectivity

7. Radars

The next file, RDLIB, contains the radar data, beginning with three header lines for identification of variables. Because of the multifunction character of the radars, the number of data lines for each equipment depends on the equipment. The first line always contains four items, viz:

RDLTYP	Radar type
RDLNAME	Radar name. See ACLIB for conditions on string.
RDLNFUNC	Number of functions performed by this radar.
RDLFUNC	4-char string giving functions performed by radar. Meaningful characters are A, G, H, I, L, Q, T, W.

The input for each function of each radar is on two lines. The first presents radar electronic data, as follows:

RDLERP	Effective radiated power (W)
RDLLFRQ	Lower frequency limit of radar type (MHz)
RDLUFRQ	Upper frequency limit of radar type (MHz)
RDLGAN	Antenna on-axis gain (dB)
RDLNBW	Receiver bandwidth (MHz)
RDLSCN	For L,H, or A search radars, scan period (sec) For T radars, lock-on and settling time (sec)
RDLTMAX	Maximum time radar will hold aircraft data (sec)
RDLMAX	Maximum range of radar against a 1 sqm target (nmi)
RDLLOS	Loss in RF lines (dB, positive value = loss, e.g., 10 means 10 dB loss)
RDLPW	Pulse width (microseconds)
RDLPRF	Pulse repetition frequency (pulses per second)
RDLNHB	Horizontal beamwidth between half-power points (degrees)
RDLNVB	Vertical beamwidth between half-power points (degrees)
RDLBL	Back-lobe level (dB)
RDLNSTACK	Number of stacked beams. Present only for A, H, L, and W functions.
RDLSMTYP	Type of SAM associated with radar

inclusion of repeater decoys. A file modification adds to the end of the first line the variable JLGAIN, the gain of the jammer if it were used as a repeater. For most jammers, this variable is not used and is set to unity. Also, a recent modification adds at the end of the data line for each band the variable JLBREL, the reliability of that band.

6. Terrain Parameters

Next consider the new file MCLUT. This contains the properties of the ground used for clutter, multipath, and optical reflectivity. After a descriptive header, the data for each terrain type are listed on its single line. For the interpretation of the variables see the analysis. The sequence of variables is as follows:

TRPTYPE	Index, terrain type
TRPCODE	Terrain type description (5-character code)
TRPA	Parameter A in terrain radar cross-section
TRPB	Parameter B in terrain radar cross-section
TRPC	Parameter C in terrain radar cross-section
TRPD	Parameter D in terrain radar cross-section
TRPROUGH	RMS roughness parameter (cm)
TRPEPSR(0)	Real part of dielectric coefficient (dry)
TRPEPSI(0)	Imaginary part of dielectric coeff (dry)
TRPEPSR(1)	Real part of dielectric coefficient (wet)
TRPEPSI(1)	Imaginary part of dielectric coeff (wet)
TRPREFL	Visible reflectivity

7. Radars

The next file, RDLIB, contains the radar data, beginning with three header lines for identification of variables. Because of the multifunction character of the radars, the number of data lines for each equipment depends on the equipment. The first line always contains four items, viz:

	For L, H, or A radars, set RDLSMTYP = 0
RDLPOL	Polarization (H, V)

The second line for each radar gives its susceptibility to six types of jamming technique and its vulnerability to five types of anti-radiation missiles. The susceptibilities are in the order from most susceptible to least, and the variable that appears is the technique. Thus the line contains:

RDLJTECH(0),... RDLJTECH(5)	Technique to which radar is susceptible; see JLRDTECH in JMLIB for specifics
RDLVUL(0),..., RDLVUL(4)	If there are fewer than 5 ARM types in ARLIB, put 0 for RDLVUL for the nonexistent types.

In addition, for the tracking function of T radars, the variable RTSCVD, the subclutter visibility, appears at the end of the second line.

For each radar, the several functions must be represented. Thus, a radar whose value of RDLFUNC is WATx will have three pairs of data lines, with values corresponding to the functions W (backup search), A (acquisition), and T (tracking), in that order. The radars are not separated by blank lines.

If the radar is an IR system (RDCLAS = Q) or an optically aimed gun (RDCLAS = G), or the function is I, the line is read with the same format, but the entries correspond to different variables. For an illumination function, the first line contains the following variables:

RIERP	Effective radiated power (W)
RILFRQ	Lower frequency limit of radar type (MHz)
RIUFRQ	Upper frequency limit of radar type (MHz)
RIGAN	Antenna on-axis gain (dB)
RINBW	Receiver bandwidth (MHz)
SKKSERV	Locke's K parameter
SKNSERV	Locke's N parameter
SKTYPE	Seeker type, 0 for pulse doppler, 1 for CW

RDLLOSS	Loss in connectors and cables (dB) in seeker receiver
SKWSERV	Locke's W parameter
SKTHRESH	Seeker threshold for monopulse system
SKNHB	Seeker horizontal beamwidth (deg)
SKNVB	Seeker vertical beamwidth (deg)
SKBL	Seeker backlobe level (dB)
RDLSMTYP	SAM type (placeholder)
RDLPOL	Polarization (H, V)

The second line for illuminators is also modified. The jamming techniques occupy the first six places, as before, but the five vulnerability values are replaced by the variables:

SKDELTAF	Doppler bank filter width (Hz)
SKATTNMX	Maximum attenuation of doppler filter (dB)
SKDFILTW	MTI filter width (Hz)
SKROLOFF	Rate of attenuation drop-off in filter bank
SKCLLEN	Length of clutter element (nmi)

IR seekers and optically aimed guns use the same input format as radars and as each other, but the variables have still other meanings. For an IR seeker, the first line contains the variables:

SKIRTYP	Short or long wavelength band
SKFOV	IR seeker field of view (deg)
SKXNEI	IR noise equivalent input (watts)
SKLRATL	Minimum S/N for seeker motion (dB)
SKURATL	S/N required for maximum seeker motion (dB)
RDLSCN	Scan period (sec)
RDLTMAX	Maximum time target held by jammed radar (sec)
SKTHRESH	Seeker threshhold for IR system
SKRATLMX	Maximum rate of seeker motion (deg/sec)
SKFLRJCTPROB	Probability of rejecting a flare

SKFLRJCTTIME	Time for decision to reject flare (sec)
SKTDWELL	Time optical system dwells on target (sec)
SKFOVO	Optical field of view (deg)
RDLSMTYP	SAM type

The second line for IR seekers uses only four variables of the 11 available, the others being merely format placeholders. The relevant variables are in the positions of the first technique value and the first three vulnerability values, and are as follows:

SKAUTON	Flag. 1 if autonomous
SKMAGNIFY	Magnification of optical system
RDSECTWIDTH(1)	Half-width of cued sector coverage (deg)
SKAIMERR	RMS aiming error for guns (deg)

For an optically aimed gun, the only meaningful variables are the last three on line 1 and the four on line 2. All others are placeholders.

8. Surface-to-Air Missiles

The next file, SMLIB, is the most complicated, since the surface-to-air missiles require the most input data. The file begins with eight header lines to identify the many SAM variables. Then for each SAM there is a descriptive line to identify the SAM, followed by six to eight data lines, the number depending on the data assembled on the effectivness of various jammers against that SAM. The input for the first line is:

SLTYP	Number identifying SAM type	
SLGUICL	Guidance class	
	A	Artillery, unguided
	C	Command
	I	Infrared
	S	Semiactive
SLNSUSC(0),	Number of jammers with effectiveness data against	

SLNSUSC(1)	this SAM type. Data averaged over all ranges for argument 0, data with a signal/noise threshold for argument 1.
SLNRT	Number of SAMs available at the launcher
SLNSTO	Number of SAMs in storage at the launcher
SLSYSREL	Overall reliability of the SAM system
SLMOBLTY	Probability the SAM is at a mobile site
SLLREL	Launch reliability
SLWREL	Weapon reliability
SLKRMIN	Miss distance inside which kill probability = weapons reliability (ft)
SLKRMAX	Miss distance outside which kill probability = 0 (ft)
SLCEP	Random error in guidance system (ft) For guns the error is angular dispersion (deg)
SLTRCT(0), SLTRCT(1)	Reaction time of SAM system for first firing at target (sec). Values are upper and lower limits of distribution.
SLTREF(0), SLTREF(1)	Reaction time of SAM system on subsequent firings (sec). Values are upper and lower limits of distribution.
SLTRLD	Reloading time of SAM launcher (sec)

The second line contains the variables:

SLHMIN	Minimum operating altitude (ft)
SLHMAX	Maximum operating altitude (ft)
SLRMIN	Minimum operating range (nmi)
SLRMAX	Maximum operating range (nmi)
SLTBECO	Time of booster cutoff (sec)
SLRBECO	Range of SAM at booster cutoff (nmi)
SLTINTER	Duration of interstage interval (sec)
SLTSECO	Time of sustainer cutoff (sec)
SLRSECO	Range of SAM at sustainer cutoff (nmi). This

	may require calculation by the user, most simply by approximating the sustainer trajectory by a straight line at constant velocity
SLTMAX	Maximum operating time of SAM (sec)
SLTIMCON	Time constant approximating SAM dynamics (sec)
SLGUIGAN	Gain of the guidance loop
SLGEELIM	Maximum transverse acceleration (g's)

The third contains propulsion data and uses metric units:

SLLOFT	Initial loft angle of SAM (degrees)
SLAREF	Reference area for SAM (m^2)
SLTHRUST(0)	Thrust of booster stage (Newtons)
SLMASSO(0)	Initial mass of SAM (kg)
SLMASSRT(0)	Rate of mass loss booster stage (kg/sec)
SLTHRUST(2)	Thrust of sustainer stage (Newtons)
SLMASSO(2)	Initial mass of sustainer stage (kg)
SLMASSRT(2)	Rate of mass loss of sustainer stage (kg/sec)
SLMASSO(3)	Initial mass of final or coasting stage (kg)

The next three lines of SMLIB pertain to the drag coefficient. For details see the analysis section under UPDSM. Line 4 contains the transition values SLMACHT(K,L), where K is the stage, L the index, and SLMACHT is the Mach number at which transition take place. Line 5 contains the numerator drag coefficient SLCD1(K,L), line 6 the denominator coefficient SLCD2, where we repeat the drag coefficient equation

$$CD = SLCD1(K,L)/(1 + SLCD2(K,L)*(Mach.No)^2)$$

Thus, line 4 reads

```
SLMACHT(0,0),SLMACHT(0,1),SLMACHT(0,2),SLMACHT(2,0),SLMACHT(2,1),
SLMACHT(2,2),SLMACHT(3,0),SLMACHT(3,1),SLMACHT(3,2)
```

and lines 5 and 6 contain the corresponding coefficients in the same order.

The next line contains the jamming effectiveness data averaged over all field test engagements. For each jammer there is a pair of numbers--the first the jammer type SLJAMTYP, the second the factor of reduction in kill probability SLJAMSUSC. For the following and last line for this SAM type, for each jammer type there is a triplet of values, the first two as in the preceding line, the third the critical signal/noise ratio SLJSCRIT required to achieve that effectiveness. The number of pairs or triplets in these lines is the appropriate value of SLNSUSC from the first data line. If there are no data, SLNSUSC = 0, the corresponding line will be absent from the library.

9. Warning Receivers

The last library file, WRLIB, contains the data on warning receivers. For each receiver type there is a descriptive header line listing the variables and the receiver name. Then the first data line contains

WLTYP	Number identifying receiver type
WLNBAND	Number of frequency bands in receiver

There follow WLNBAND lines giving the electronic properties per band. WLNBAND cannot exceed 20. For each band, the line includes,

WLBNUM	Number identifying band
WLBCODE	3-character code describing band, e.g., MID
WLBLFRQ	Lower frequency limit of band (MHz)
WLBUFRQ	Upper frequency limit of band (MHz)
WLBGAN	Gain of receiver antenna (dB)
WLBSENS	Sensitivity of receiver (decibels below 1 milliwatt = dBm)

After these NWLBAND lines is a single line with the parameters of the "rosette" antenna:

WLNLEAF	Number of leaves in rosette
WLBS	Angle from nose of center of first leaf (deg)
WLDIP	Angle of elevation of antenna system from horizontal (positive upward) (deg)
WLNHB	Horizontal beamwidth between half-power points (deg)
WLNVB	Vertical beamwidth between half-powers (deg)

Except for the calculation of the parameters of the drag coefficient of the SAMs, the preparation of these library files only requires collection of data. RJARS can be run even if there is only one equipment in each file.

PREPARATION OF SIMULATION FILES

The preparation of the simulation files will usually require detailed investigation. These files are:

1. TERRA	Terrain heights
2. ACVIS	Aircraft visibility over terrain (input)
3. ACSGT	Aircraft visibility over terrain (output)
4. RIDGE	Radar-terrain ridge file (input)
5. RDRDG	Radar-terrain ridge file (output)
6. BLUMX	Flight path - fixed wing aircraft
7. CHAMP	Flight path - helicopters
8. SCENA	Simulation run parameters
9. DISPL	Graphics output file

The file TERRA contains the detailed terrain data, giving for each terrain point the coordinates (rectangular or global) and the altitude in feet above sea level. ACVIS displays the intervals of visibility from each radar to each aircraft. It may be prepared from maps or by running the simulator in mode VIS = 2 (see below). Mode VIS = 2 creates the file ACSGT from the aircraft flight paths, radar positions, and terrain file TERRA. ACVIS may then be transcribed from ACSGT.

Similarly, RIDGE presents the ridge data as a function of azimuth from each radar. It may be prepared by the RJARS simulator by running the mode VIS = 4, which will produce the output file RDRDG. RIDGE may then be transcribed from RDRDG. At RAND, both ACVIS and RIDGE are usually prepared by CAGIS, which is considerably more efficient than RJARS at this type of calculation.

The files BLUMX and CHAMP contain the flight path data. They are prepared at RAND by using the appropriate flight path simulator (Blue Max or Champ). An outside user, who does not have Blue Max or Champ, can use his or her own simulator provided the output format matches the description below. The file DISPL is an output file that contains the data necessary for RAND's graphics program. It is in the format employed by CAGIS, and, as mentioned, this part of RJARS is the only part not fully portable.

The scenario file SCENA contains the information required for each simulation run. It includes number of elements (aircraft, radars, etc.), a set of flags describing the simulation conditions, choices for general parameters such as time step, printing options, the details on aircraft types, flight paths, and maneuvers, radar types, positions, and reporting, site and command site locations and reporting, and communications delays.

1. Terrain Data

The file TERRA has been described in the subsection UPDTR of the analysis section. After a single header line, the following items appear on a single line:

TRLGMIN	Minimum terrain longitude
TRLGMAX	Maximum terrain longitude
TRLTMIN	Minimum terrain latitude
TRLTMAX	Maximum terrain latitude

TRDX	Longitude element length
TRDY	Latitude element length

After another header, each line of TERRA contains the latitude, longitude, height above sea level, and type of two terrain elements in the order

TRLAT(1),TRLONG(1),TRALT(1),TRTYP(1),
TRLAT(2),TRLONG(2),TRALT(2),TRTYP(2),

Entries in TERRA can be either in RJARS rectangular coordinates or in global coordinates (degrees and fractions of degrees, not degrees, minutes, and seconds), but then the matching coordinates must be used when running the simulation. The available terrain types are described in the discussion of MCLUT.

The terrain field must be organized such that the number of elements along each latitude or longitude line is odd. It may be scaled to any size. Elements may be entered in any order. After all have been entered, the total number being odd, an additional element should be used as a terminator. This terminator should have its latitude and longitude outside the terrain field and an altitude greater than 30,000 feet, thereby corresponding to no point on Earth. RJARS will recognize this fictitious element as the end of the file.

2. Aircraft/Radar Intervisibility Data

The mode of operation of terrain simulation is controlled by the parameter VIS, which is an input from the scenario file SCENA. If the terrain modes VIS = 0 or VIS = 3 are to be run, it does not matter what is in the files ACVIS and ACSGT. If VIS = 2 is run, it will generate the file ACSGT. The items appearing on a line are as follows:

NTAB	Line number
TON	Time aircraft becomes visible to radar

TOFF	Time aircraft ceases being visible to radar
IAC	Number identifying aircraft
IRD	Number identifying radar

In the file ACSGT, the items appear in the order of increasing values of TON. If several aircraft become visible to several radars at the same time, RJARS will sequence them by aircraft number and then by radar number. The end of the file is marked by the terminator line

```
1000,9999,0,1,1,
```

The first element may be any number larger than the last number used to count data lines. The important item is the 9999, which ensures that RJARS will not attempt to read past the end of the file during the simulation. After VIS = 2 has been run, the file ACSGT should be copied to another name for preservation, since ACSGT is destroyed each time RJARS is run. As mentioned, at RAND ACVIS is usually prepared by CAGIS, which organizes the radars, flies the aircraft, and produces the output file in the indicated format.

3. Ridge Data

The associated files RDRDG and RIDGE contain the ridge data. The data are used for clutter calculations to determine if the ground is visible at relevant points, and for optical equipments to ascertain if the background is sky or ground. RJARS prepares the ridge data by running it with VIS = 4, which produces the file RDRDG. Like ACSGT, RDRDG is destroyed each time RJARS is run, so it should be transcribed immediately.

After a header line, RDRDG contains a list of radars, with each line containing the radar identification, latitude, and longitude. These are in the appropriate coordinate system (rectangular or global). After the listing and another header comes the main part of the ridge file, arranged so all the data for a given radar are in sequence. On each data line there are five items as follows:

RDIDENT	Radar identification
AZIMUTH	The angle for which ridges are being calculated (deg)
RDRIDGDPR	Depression angle of the line of sight to the present or previous ridge (deg)
RDRIDGDIST	Distance to the ridge or ground intercept point (nmi)
TYPE	Integer setting if we are seeing an entry into ground visibility (0) or a masking (1)

For each radar and each direction, the first item represents the point under the radar, for which the depression angle is 90 degrees, the distance is 0 nmi, and the type is 0, indicating the ground is visible. The next line will be the first point where the ground becomes masked, with the depression angle to that point, the range to that point, and the type 1. If the ground rises enough beyond this point that it again becomes visible to the radar, the depression angle is that of the first ridge, the range is to the new point, and the type is 0. This sequence is continued out to the terrain border in the given direction. If the ground is visible at that location, an artificial ridge is created, so RJARS radars will not see outside the terrain limits. Clearly, the number of ridges in a given direction is arbitrary. Just like ACVIS, at RAND RIDGE is usually prepared by CAGIS.

The number of directions in which ridges are calculated is controlled by the parameter nrdgangle, which is located in a #define statement at the beginning of the assemblage of variable declarations rjarsdecls.h. It is defaulted to 37 in both RJARS and CAGIS, corresponding to a 10 degree spacing between ridges. If the user wishes to change the ridge spacing, it is necessary to change nrdgangle and then recompile RJARS.

4. Scenario

The scenario file SCENA contains the information required for the execution of a simulation. It must be prepared for each run, although sometimes only a few changes need be made to modify SCENA for new conditions. After a header, the first line of SCENA contains the variables:

NAC	Number of aircraft
NRD	Number of "radars" (includes long-range search, height-finders, acquisition radars, trackers, multifunction radars, IR systems, and optically aimed guns.
NSI	Number of sites
NSC	Number of command sites
NREP	Number of Monte Carlo repetitions
VIS	Over-terrain visibility mode 0 No terrain 1 Visibility from ACVIS file 2 Preparation of visibility file 3 Terrain included directly 4 Preparation of ridges
NDOF	Number of degrees of freedom for aircraft rcs 3 Radar cross section data in azimuth plane only 5 Radar cross section data in both azimuth and elevation planes

After another header line to indicate the variables that follow, the next line of SCENA includes the following set of simulation flags. Unless otherwise stated, for each flag 1 = Yes, 0 = No.

IAVAIL	Recalculation of radar availability 1 Availability recalculated at each iteration 0 Availability calculated only at beginning of simulation
ICLUT	Clutter calculations included
IMULT	Multipath calculations included
IWET	Wet terrain properties
ISNOW	Terrain covered with snow
IJANUS	JANUS probability calculations at end of simulation
IMSFD	Permanent radar numbers from threat laydown

	included in radar input data
INIGHT	Night operations. IR and optical systems disabled.
ILOBS	Special calculations for low-observable aircraft in effect

Following another header are four more flags pertaining to optical systems:

IOPT	Optically aimed weapons employed	
ISUMMER	Contrast transmission for season	
	1	Summer conditions
	0	Winter conditions
ISEEING	Contrast transmission conditions	
	1	Median seeing
	0	Seeing quality exceeded only 10 percent of the time
ISUN	Sun visible in sky	

After another header is a set of general simulation variables:

DT	Time step (sec)
TFIN	Termination time of scenario (sec)
DTG	Guidance time step (sec)
TDAY	Time of day (hr)
SEED	Integer to start the random number generator

Another header, then a set of character variables:

SIZE	Character to indicate number of change points in aircraft flight maneuvers. Use S if the number of change points is 10 or fewer, M if 11-25, L if greater than 25
SRBMOD	Character for tracker servo mode. F, fixed servo bandwidth; A, adaptive servo bandwidth
PRNTMOD	Character for radar printout. S, shortened format; anything else, full descriptive

	format
SMPRNT	Character for SAM printout. Y, prints details of flight under control of IPG; N, prints details only when range to target < 1000 ft
HF	Character for height-finders. N, height finders not included
NOSHT	Character for SAM operation. Y, SAMs never shoot
RCTDISTRIB	Character for the type of distribution for the SAM reaction and refire times. U, uniform; N, normal

Another header, then printing controls and miscellaneous parameters:

IP	Control. Search radar printout occurs each IP'th time step
IPA	Control. Aircraft position printout occurs each IPA'th time stop
IPG	Control. SAM detailed data printout occurs each IPG'th subdivided time step

COORD	Input-output coordinate choice	
	1	Input and output rectangular
	2	Input and output global
	3	Input rectangular output global
	4	Input global output rectangular

PRBTYP	Probability choice or jamming table	
	0	Sinusoidal curve fit
	1	"Cookie cutter"
	2	Jamming effectiveness tables, averaged over jamming conditions
	3	Jamming effectiveness tables, signal/noise ratio as parameter

The next two lines of SCENA are present only if COORD > 1, so either input or output is global; first a header, then on one line:

DLATO	Degrees for reference latitude
MLATO	Minutes for reference latitude
SLATO	Seconds for reference latitude
HLATO	Hemisphere for reference latitude (N,S)

then another header, then on one line:

DLONGO	Degrees for reference longitude
NLONGO	Minutes for reference longitude
SLONGO	Seconds for reference longitude
HLONGO	Hemisphere for reference longitde (E,W)

Following another header are the parameters for the field of action of the simulation in rectangular coordinates.

LATMIN	Minimum ordinate (nmi)
LATMAX	Maximum ordinate (nmi)
LONGMIN	Minimum abscissa (nmi)
LONGMAX	Maximum abscissa (nmi)

After another header SCENA presents the initial parameters for each aircraft. If COORD = 2 or COORD = 4, then on each line are the variables:

ACIDENT	Identification number of aircraft
ACTYP	Number identifying aircraft type
ACGROUP	Group with which the aircraft is initially associated
ACDLAT	Degrees of initial latitude
ACMLAT	Minutes of initial latitude
ACSLAT	Seconds of initial latitude

ACHLAT	Hemisphere of initial latitude (N,S)
ACDLONG	Degrees of initial longitude
ACMLONG	Minutes of initial longitude
ACSLONG	Seconds of initial longitude
ACHLONG	Hemisphere of initial longitude (E,W)
ACZALT	Initial altitude above terrain (ft)
ACHDG	Initial heading (degrees) (clockwise from north)
ACVEL	Initial velocity (knots)
ACJMTYP	Type of jammer on board the aircraft
ACWRTYP	Type of warning receiver on board the aircraft
ACRDJM	4-character code--types of radars the aircraft should jam. L, H ,A, T, or I represent possible choices, 0 fills out the code to four characters
ARMN	Number of ARMs on board
ACPAINTL	Color of lower surface
ACPAINTU	Color of upper surface

If COORD = 1 or COORD = 3, the variables ACDLAT,...ACHLONG are absent, and the variables

ACLAT	Initial ordinate (nmi)
ACLONG	Initial abscissa (nmi)

appear between ACGROUP and ACALT.

If ARMN > 0, the next line of SCENA contains the input data for all the ARMS in that aircraft, viz;

ARMTYP(J,0)	Type of first ARM on aircraft J
ARMBD(J,0)	Frequency band of first ARM on aircraft
ARMRD(J,0)	Target radar of first ARM on aircraft J. If the ARM is preprogrammed, there will be a radar number here. If the ARM is opportunistic, the value 0 will appear
ARMTYP(J,1)	Type of second ARM on aircraft J, etc.

The aircraft data and ARM data if present are listed for all NAC aircraft in the simulation.

After the next header the flight man uvers for the number of aircraft (NAC) and the communications connecting and cutting for the fictitious NAC+1st aircraft are listed. Beginning with aircraft number 1, the maneuvers for each aircraft are listed in nondecreasing time sequence, in the form:

TIME	Time when maneuver occurs
CHANGE	5-character code identifying maneuver type. For details see the analysis section
X	First parameter for maneuver
Y	Second parameter for maneuver

The last maneuver for each aircraft and for the communications system must have TIME = 9999.

After another header comes the radar parameters. If COORD = 1 or COORD = 3, these are:

RDIDENT	Identification number of radar
RDMSFDNO	If called by IMSFD (see above) number of radar in threat laydown. This number must be followed by a space before the comma.
RDLAT	Radar ordinate (nmi)
RDLONG	Radar abscissa (nmi)
RDZALT	Radar height above terrain (ft). Includes antenna height.
RDTYP	4-character code identifying radar type.
RDATSITE	Site to which radar reports
RDSECTMEAN	Center line of radar sector of coverage (deg)
RDSECTWIDTH	Half-width of sector of coverage (deg)

If COORD = 2 or COORD = 4, the ordinate and abscissa are replaced by degrees, minutes, seconds, hemisphere as with aircraft. All NRD radars must be listed. They may be in any order. IR and optical systems are subsumed under the category "radar".

After another header comes the single item DLYMULT. This is a factor that multiplies all the communications and decision delays in the system. It would normally be defaulted to 1.

Following another header is the site data. For all NSI sites, the parameters on the site line if COORD = 1 or 3 are:

SIIDENT	Site identification number
SILAT	Site latitude (nmi)
SILONG	Site longitude (nmi)
SIZALT	Site height above terrain (ft)
SIUPDLY	Communications delay, site to command site (sec)
SIDNDLY	Communications delay, command site to site (sec)
SIRPT	Command site to which the site reports
SIMOBILE	Flag. 1 - Site is mobile and subject to availability restrictions. 0 - Immobile, available

If COORD = 2 or 4, the latitude and longitude in rectangular coordinates are replaced by their values in degrees, minutes, seconds, and hemisphere.

Finally, after still another header, SCENA is completed with the command site data. For each of NSC command sites, with COORD = 1 or 3:

CSIDENT	Command site identification number
CSLAT	Command site latitude (nmi)
CSLONG	Command site longitude (nmi)
CSZALT	Command site height above terrain (ft)
CSDECDLY	Command site decision delay (sec)

For COORD = 2 or 4, see above. This concludes the description of the input files.

OUTPUTS

RJARS provides outputs at each time step, summaries at the end of each run, and a graphics output file. When several Monte Carlo runs are executed, the detailed outputs (position, tracking error, etc.) are printed only during the first run. Thereafter, only events (detections, SAM launches, etc.) and the summaries are printed.

1. Printed Outputs

Aircraft outputs appear each IPA'th time step, and may be in rectangular or global coordinates. Specific items printed are latitude, longitude, altitude, heading, velocity, and the number of ARMs (anti-radiation missiles) remaining on that aircraft. If an aircraft is maneuvering, the maneuver state (e.g., current heading) is printed every time step during the maneuver.

Search and acquisition radar outputs, printed each IP'th time step provided the target has been in the radar main beam during that time step, are time of detection, identification, signal, signal-to-noise ratio, range, bearing, probability of detection, and position measurement errors. Trackers display the same information except that the time is replaced by the word TRACK, the probability of detection by the elevation, and the errors may include the word NO, if the lock in azimuth or elevation is broken by jamming. Trackers print at every time step for which they are turned on. If clutter is being calculated, the amplitude of the clutter signal is printed after the true signal value. If multipath is included, the multipath factor (modification of the true signal) is printed. For optical systems, indicated by the word OPTICS in the print, the contrast at the viewer and the cycles across the target divided by the threshold detectibility are printed, plus a detection probability and random draw for the detection or recognition phase of operation. IR seekers have the same printout as trackers, marked by the word IRSEEK and not including the tracking errors.

ARMs in flight print at each time step their latitude, longitude, altitude, target identification, and the range to the target. If they self-destruct, it is announced. If and when an ARM reaches its target, the kill probability and outcome are presented.

SAMs in flight print under the control of the parameters SMPRNT and IPG. Normally SMPRNT is set to N to avoid excessive output, in which case at each time step for which the SAM is more than 1000 feet from its target it will print identifications, the SAM state (boosting, sustaining, coasting) and the ranges from the SAM to its launcher and its target. If SMPRNT = Y, or if the SAM is less than 1000 feet from its target, it will print at each IPG'th subdivided time step within the full time step the following variables:

SAM	SAM identification
TGT	Target identification
TIME	Current time (sec)
FLTIME	Flight time since launch (sec)
STATE	Current state (B, I, S, or C)
ACLAT	Aircraft latitude including measurement errors (nmi)
ACLONG	Aircraft longitude as above (nmi)
ACALT	Aircraft altitude as above (ft)
LAT	SAM latitude (nmi)
LONG	SAM longitude (nmi)
ALT	SAM altitude (ft)
RANGE	SAM range from tracker (nmi)
ELEV	SAM elevation angle from tracker (deg)
AZ	SAM azimuth angle from tracker (deg clockwise from North)
VEL	SAM velocity (fps)
PITCH	Elevation angle of SAM velocity (deg)
HDG	Heading angle of SAM velocity (deg clockwise from North)
RNGTM	Range from SAM to target (nmi)
ELEVTM	Elevation angle from SAM to target (deg)
AZTM	Azimuth angle from SAM to target (deg)

These quantities are printed on three lines with the words displayed as listed. If the SAM uses a semiactive seeker, the following quantities are printed on an additional line:

SEEKAZ	Azimuth toward which the seeker is pointing (deg)
SEEKEL	Elevation toward which the seeker is pointing (deg)
AZCOM	Commanded change in azimuth direction (deg)
ELCOM	Commanded change in elevation direction (deg)
AZSK	Azimuth displacement from boresight (deg)
ELSK	Elevation direction from boresight

Aircraft may defend themselves against IR systems by launching flares. At the time of launch of an IR SAM, a random draw determines if the aircraft detects the launch. The event and result are printed. If the detection was successful, the aircraft may launch a succession of flares, for which the launch event and success are printed. The seeker may or may not identify and reject each flare. At each IPG'th subdivided time step while the SAM and flares are in flight, the angles of the seeker to the target and to each flare are printed, displaying how it may transfer allegiance as time progresses. Flare burnout and destruction events are printed. SAM countdown and launch events are printed at their time of occurrence. When the SAM reaches its target, the times of closest approach, range from the launcher, miss distance, kill probability, value of the random draw variable, and outcome are displayed.

The summary printouts are described fully in the analysis section under the subroutine OUTPUT, and the user should refer to that material.

2. Graphical Outputs

The RJARS graphics output is incorporated in the file DISPL. As has been stated many times, DISPL is in CAGIS format, and cannot be used or read unless CAGIS is available. In addition, the preparation of DISPL uses the CAGIS subroutine mg_put, and if RJARS is to be used at a facility which does not have CAGIS all appearances of this subroutine must be deleted. The DISPL file contains the following information, marked by the parameter dtag, which tells CAGIS what parameters are being transmitted:

dtag = 1	Radar parameters IDENT, LAT, LONG, ZALT, TYPE, SITE, COMMAND SITE, MINIMUM SECTOR COVERAGE ANGLE, MAXIMUM SECTOR COVERAGE ANGLE
dtag = 2	Flight path parameters IDENT, TIME, LAT, LONG, ZALT, ALT, VEL, HDG
dtag = 3	Flight path extension or RJARS aircraft path parameters as in dtag = 2
dtag = 4	Site or command site parameters IDENT, LAT, LONG, ZALT, ALT, (SIRPT for sites, -1 for command sites)
dtag = 5	Initial values NAC, NRD, NSI, NSC, TRLTMAX, TRLTMIN, TRLGMAX, TRLGMIN
dtag = 6	Probability data for JANUS ACIDENT, RDIDENT (0 for killing before or after LOS1, LOSN), INTERVAL NUMBER, START OF INTERVAL, END OF INTERVAL, KILL PROBABILITY DENSITY
dtag = 7	Maneuver profile ACIDENT, PROFTIME, PROFCHANGE, PROFX, PROFY

The foregoing parameters are sent to DISPL only once (end of simulation for dtag = 6, as developed for dtag = 3, beginning of simulation for all others). In addition, information is sent on each iteration, marked by the following values of dtag :

dtag = 10*JREP + 1	Search and acquisition radar summary parameters RDIDENT, ACIDENT, CODE for event type, EVENT TIME. Code 1 for entry, 2 for detection, 3 for loss of detection, 4 for exit
dtag = 10*JREP + 2	Track radar summary parameters RDIDENT, ACIDENT, TIME ACQUISITION ON, TIME ACQUISITION OFF, TIME TRACKING ON, TIME TRACKING OFF
dtag = 10*JREP + 3	SAM summary parameters IDENT, TARGET, SAM NUMBER FROM LAUNCHER, LAUNCH TIME, INTERCEPT TIME, CODE for intercept result: 0 launch failure,

	1 still in flight, 2 lost track, 3 crashed, 4 missed target, 5 shot target down
dtag = 10*JREP + 4	Anti-radiation missile summary parameters AIRCRAFT carrying ARM IDENT, NUMBER of ARM launched from that aircraft, TARGET RADAR, LAUNCH TIME, ARRIVAL TIME (9999 if destructed in flight) CODE for result: 0 lost track, 1 missed target, 2 killed target, 3 still in flight
dtag = 10*JREP + 5	SAM flight parameters LAUNCHER IDENT, TARGET IDENT, NUMBER from that launcher, CURRENT subdivided TIME, LATITUDE, LONGITUDE, ALTITUDE above sea level
dtag = 10*JREP + 6	Radar or optical scan limits RDIDENT, TIME, SCAN LOWER LIMIT, SCAN UPPER LIMIT. Printed only upon changes
dtag = 10*JREP + 7	Jammer on and off times ACIDENT, RDIDENT, TIME, CODE for event: 0 on, 1 off

The RJARS graphics program, developed by Gail Halverson, can produce a variety of displays. Usually, the terrain is displayed first, then the defensive laydown, indicated by a set of icons at the appropriate positions. The aircraft paths are shown, and their condition with respect to the defense indicated by changing color. Thus, an undetected aircraft path is black, one that has been detected by search radars is gray. When radars are attempting to acquire, the path is blue, when they have acquired and a tracker is on it becomes yellow. If a SAM is in flight against that aircraft the path turns orange. The path of a SAM is in red. Intercepts are marked by small stars for misses, large stars for kills. If an aircraft is killed its path does not continue. Paths may be laid down all at once, or the time may be stepped. Changes in radar scan coverage or jammed condition are shown by changing marks on the radar icon. The successive iterations may be shown in sequence. The summary results are presented as a legend alongside the main display.

RJARS may be compiled with the standard UNIX C compiler. Except for the frequently mentioned display limitations, there should be no problems with program portability. To compile RJARS on another installation, remove all appearances of the command mg_put from the program.

RJARS will abort if there are data read-in errors, and indicate which type of equipment (aircraft, radar, etc.) is at fault. These errors can usually be found quickly by scanning the appropriate section of SCENA. Frequent causes of such errors are typing a period instead of a comma as a separator between data entries, or failure to use the proper type of quotation marks on character entries.

There appeared at this point in the earlier version of this report a set of sample input files and typical output. These were provided as accompaniments to the code, so the reader could test with files known to be correct. Since the code is no longer included in the report, these sample files and typical outputs have been dropped.

V. PROGRAMMER'S GUIDE

RJARS is a completely sequential program. It contains no goto statements, but tests flags or variables and either performs or doesn't perform the indicated program sequences. Nesting levels may be very deep (level 13 is reached in UPDRS), but no loop is exited with a branch command, so the sequence should be easy to follow by simply counting DO and END statements in the PL-I version, or { and } in the C version.

The only unusual program device in the PL-I version of RJARS is used during the reading of the library files, and is caused by the one-way nature of the PL-I input stream. To illustrate how it operates, consider the input of the warning receiver data. Before these data are read, the type of receiver on each aircraft has been read from SCENA. A parameter K is set equal to the number of aircraft, and the command DO while K ≠ 0 is given. The data on the first type of warning receiver in the library are read from WRLIB. The sequence of aircraft is run through. For each aircraft whose designated warning receiver is of the first type, the parameters of the receiver on that aircraft are set equal to the library parameters, and K is decremented. After all aircraft have been checked, data on the second receiver are read and the process repeated. The operation will terminate either when K = 0 (all aircraft receivers properly loaded) or an end-of-file condition is reached. In the former case the program continues, in the latter the flag EOFFLG is set to 4. All the library files are read using this device. If EOFFLG has any value other than zero at the end of the data read-in, RJARS aborts and prints the proper error message.

RJARS has been translated into the C computing language to operate on UNIX installations. There are only a few significant differences between the programs. The separation of the radars into two lists, which was implemented in the PL-I version to save space, was not included in the C version, since our installation had plenty of computing space available. The arraying structure in C is different from PL-I. First, in C the first element in an array is designated with the index value 0, whereas in PL-I the index is 1. Consequently, the arrays had to be carefully tracked, and frequently 1 had to be subtracted from an input index value, to be certain that array elements are stored in the proper location. Second, array structures in C require full qualification of their variables, which leads to such equivalences as:

ARMALT(J,L) → ANTIRADIATION_MISSILE [J].ARM [L].ALT

where the first expression on the right denotes an outer structure, ARM is an inner structure, and ALT is the variable. Because of this requirement the C program has longer statements for the same content.

The terrain indexing is different in the C program. Since matrices in C must begin at 0, and the terrain matrix for PL-I has been centered at 0,0, the parameters ntr,ntr2 have been added to the externally defined matrix sizes that are required in the C program. Here ntr is the full size of the terrain matrix, and ntr2 is half that. The terrain elements are centered on the point ntr2,ntr2, and the index finder subroutine TRINDEX has been adjusted to reflect this.

There are command and notation differences between PL-I and C, such as PUT EDIT in PL-I is replaced by printf in C. These have all been attended to in the translation. The equations which have been programmed are identical, so aside from the indicated differences the programs are quite equivalent.

The C language includes a rewind command for files, so the special read device described above for PL-I is not necessary. Indeed, certain of the files are rewound of necessity, such as the rewind of ACVIS for Monte Carlo sequencing. Also, some files are read twice during data input to first establish the amount of space that must be allocated and then read in the data.

The major programming difference between the present version of RJARS and the C program at the time the previous report was written is the use of dynamic allocation. The dynamic allocation procedures were developed by Jim Gillogly and then extended by the author. The problem with the earlier version was that all arrays had their dimensions set by numbers defined at compile time. Thus, if the user ever desired to treat a scenario with 500 radars, it would be necessary to provide space for 500 radars for all scenaria. This was an extremely wasteful process.

The cure was to define all arrays that might have variable dimension by pointers. For example, the aircraft array is defined as *AIRCRAFT. The number of aircraft NAC is read from the scenario. Then the following command:

```
if ((AIRCRAFT = (struct acstruct *) calloc(NAC,
    sizeof(struct acstruct))) == NULL) outofmem("AIRCRAFT");
```

allocates space for NAC aircraft, each having all the variables defined in the structure acstruct. If there is insufficient space available, the subroutine outofmem will indicate that the space overflow occurred during the allocation to the array AIRCRAFT. Similar declarations are used for all such arrays. Those declarations that involve only the dimensions NAC, NRD, NSI, NSC, NCP, or NGUI--respectively the number of aircraft, radars, sites, command sites, change points, or guidance time steps--are collected in the subroutine allocate. Others, such as those that depend on the number of ridges in a given direction from a radar, are allocated after the necessary information to scale them has been read or calculated.

The space savings produced by this dynamic allocation can be very great. Thus, suppose the user with 500 radars discovers that only 20 of them are involved in the actual simulation, and that the rest are present but never doing anything. The 20 active radars can be abstracted from the set and excursions from the base case can be run using only about 4 percent of the space previously used. In contrast, with the older version of RJARS all 500 radars would have to be kept for the excursions.

Another important change is the addition of a graphics program. This is directly associated with the CAGIS system and is the only portion of RJARS that is not portable. The graphics program was developed by Gail Halverson. Much of it is resident in CAGIS. The only graphics function in RJARS is the writing of a file DISPL, that contains the data which will be presented in the graphics. The transmitted information uses the integer dtag to identify which graphics data are being sent, the integer dtype to show its nature (only dtype = 7, corresponding to floating point numbers, is used), and dnitems to show how many items are on a data line. The array dbuf, containing 20 elements, carries the actual numbers. If a variable which is to be sent to DISPL is not a floating point number, it must be cast as one. The command mg_put(dtag, dtype, dnitems, dbuf) calls a CAGIS routine to convert the items to a compressed format and store them for later use by the CAGIS graphics procedures.

Among the information items sent to DISPL are the following:

1. Numbers of aircraft, radars, sites, command sites, and terrain limits
2. Radar identification, type, location, and site
3. Site identification and command
4. Command site location
5. Flight path identification with position and velocity
6. Aircraft continuation of flight path position and velocity
7. Scan limits for sector scanning systems
8. Flight paths of SAMs
9. Times when aircraft enters, is detected, or leaves search radar field.
10. Times of assignment, acquisition on and off, and tracking on and off for a track radar.
11. Results of SAM endgames
12. Results of ARM endgames

The difference between flight path and flight path continuation is that if the aircraft flight path has been derived from a flight path generator, then it may also involve straight and level flight extensions at the beginning and end of the path. These may vary from iteration to iteration depending on how long the aircraft survives. The longest continuation is used for the graphics.

Certain of these items, such as the radar locations, are fixed throughout the simulation and need be transmitted only once. Others, such as the results of SAM endgames, may differ and have to be sent every iteration. The iteration dependent information is marked by the tag, which at each iteration is incremented by 10 for the appropriate variables.

GLOSSARY

RJARS involves so many variables that a glossary is mandatory. Most of the variables are themselves arrays and are included in structures. The dimensions of the arrays, a description of the variables, and their units should be given. To avoid much writing we use the convention that the first argument of the array is subsumed in the variable name. Thus, the first variable in the list, ACACCL, belongs to the structure AIRCRAFT, as indicated by the initial letters AC, and is a one-dimensional array of size NAC. The structures, size, and opening letters of the variables are as indicated.

AIRCRAFT (NAC)	AC
ACSTATE (NAC)	AS
ANTIRAD_MISSILE (NAC)	ARM
ANTIRAD_MISSILE_LIB	ARL
COMMAND_SITE (NSC)	CS
DETECTIONS (NRD,NAC)	DT
DETSUM (NRD,NAC,10)	DTS
FACET (NFACET)	FA
FLARE (NFLARE)	FL
FLIGHT_PATH (NFLPATH)	FP

IRDATA (variable)	IR
JAM (NAC)	JM
JAMMER_LIB	JL
PROFILE (NAC+1)	PROF
RADARS (NRD)	RD
ILLUMINATION_RADAR(NRD)	RI
SEARCH_RADAR (NRD)	RS
TRACK_RADAR (NRD)	RT
RADAR_LIB	RDL
SITE (NSI)	SI
SEEKER (NRD)	SK
SIMPLE_SAM (NRD)	SM
SAMLIB	SL
TERRAIN (2*NX + 3, 2*NY + 3)	TR
TERRAIN_PARAMETERS (nttype)	TRP
VISDATA (NAC,NRD)	VIS
WARNING_RECEIVER (NAC)	WR
WARNING_RECEIVER_LIB	WL

The words aircraft, radar, etc. will be omitted from the description of the variable, since they are implicit in its designation. Variables not in the structures are described in full. Variables that are part of multidimensional arrays will have only those dimensions beyond the first indicated. For example, the variable RTLATI, which has the dimensions NRT, NGUI, is referred to in the glossary as RTLATI (NGUI), the first dimension NRT being implicit in the initial RT.

GLOSSARY OF VARIABLES

Variables with the same meaning but different references are listed together. Flag status if set is shown. Matrix dimensions exclude first if identified by variable name. Upper and lower case letters for variables are distinguished.

ACACACCL	Acceleration (knots/sec)
ACACCLMB	Climb rate (fps)
ACACPITCH	Pitch acceleration (deg/sec2)
ACACPITCH1	during first part of climb maneuver
ACACPITCH2	during last part of climb maneuver
ACACTURN	Turning rate (deg/sec)
ACALPHA1(RCSNAZ)	Radar cross-section, 3 deg of freedom (dB/1sqm)
ACALPHA2(RCSNAZ,RCSNEL)	Radar cross-section, 5 deg of freedom (dB/1sqm)
ACALT	Altitude above sea level (ft)
ACALTG(NGUI)	true interpolated
ACALTO	initial
ACALTP(3)	present and two past values
ACALTPROJ	projected from SAM launch time to present
ACBANK	Bank angle (+ right wing down) (deg)
ACBANKG(NGUI)	true interpolated
ACBANKO	initial
ACBLKSTRT	Time of starting of blink jamming (sec)
ACC	Acceleration entry for SRVERR
ACCEL	Subroutine for aircraft acceleration
ACCRASHT(20)	Time of apparent aircraft crash (sec)
ACDECNUM(10)	Identification number for decoys
ACDELALTPROJ	Change in projected altitude in DTG
ACDELLATPROJ	latitude
ACDELLONGPROJ	longitude
ACDESTT	Time of arrival at destination
ACDISP	Displacement of towed decoy from parent (ft)
ACDLAT	Degrees of latitude
ACDLONG	Degrees of longitude
ACEGLNLG	Effective glint length (ft)
ACFLAZ(2)	Azimuth of each flare launcher from forward
ACFLFUSDISP	Longitudinal displacement of flare launchers from center of nose-to-tail line (ft)
ACFLTILT	Tilt of flare launcher from vertical (deg)
ACFLWINGDISP(2)	Lateral displacement of each flare launcher from center line (ft)
ACFPPATHDLY	Time of joining flight path (sec)
ACFUSDIAM	Fuselage diameter (ft)
ACGLNAZ	Azimuth glint error
ACGLNEL	Elevation glint error
ACGROUP	Group to which ac belongs

ACGROUPO	Initial group to which ac belongs
ACHDG	Heading (deg clockwise from North)
ACHDGG(NGUI)	true interpolated
ACHDGO	initial
ACHDGP(3)	present and two past values
ACHLAT	Hemisphere of latitude (N,S)
ACHLONG	Hemisphere of longitude (E,W)
ACIDENT	Aircraft identification number (tail no.)
ACIRSIG(2,13,13)	Radiant intensity in each spectral band in steps of azimuth and elevation
ACJMTYP	Type of jammer on aircraft
ACKILLT	Time when aircraft killed
ACLAT	Latitude in rectangular coordinates (deg)
ACLATG(NGUI)	true interpolated
ACLATO	initial
ACLATP(3)	present and two past values
ACLATPROJ	projected from SAM launch to present time
ACLENGTH	Overall length (ft)
ACLFLARE	Number of flares per dispenser
ACLIB	File. Aircraft dimensions and parameters
ACLONG	Longitude in rectangular coordinates (deg)
ACLONGG(NGUI)	true interpolated
ACLONGO	initial
ACLONGP(3)	present and two past values
ACLONGPROJ	projected from SAM launch to present time
ACLOS1	Time of first LOS to target for JANUS (sec)
ACLOSN	Time of last LOS to target for JANUS (sec)
ACMLAT	Minutes of latitude
ACMLONG	Minutes of longitude
ACNOSEAREA	Nose projected area (sq ft)
ACNOSEDET	Probability of detecting a flare nose-on
ACNPITCH	Pitch angle of nose (deg from horizontal)
ACNPITCHG(NGUI)	true interpolated
ACNPITCHO	initial
ACPAINTL	Color of lower surface (2-char code)
ACPAINTU	Color of upper surface (2-char code)
ACPITACDN	Maximum acceleration in downward pitch (g's)
ACPITACUP	Maximum acceleration in upward pitch (g's)
ACPITCH	Pitch angle (deg from horizontal)
ACPITCHG(NGUI)	true interpolated
ACPITCHO	initial
ACPITCHP(3)	present and two past values
ACPITCHC	Critical pitch angle (deg)
ACRCS	File. Radar cross-section
ACRCSAZRES	Radar cross-section resolution in azimuth (deg)
ACRCSELRES	Radar cross-section resolution in elevation (deg)
ACRCSNAZ	Number of elements in azimuth in cross-section
ACRCSNEL	Number of elements in elevation in cross-section
ACRDJM	General types of radars to be jammed
ACRDJMCL(4)	individual
ACRDVISKEY(NRD)	Key to file visibility data

ACREFL(2)	Reflectivity of lower and upper surfaces
ACROTORRCS	Radar cross section of helicopter rotor (dB/1sqm)
ACSENSORRCS	Radar cross section of mast-mounted sensor(dB/1sqm)
ACSGT	File. Visibility data output file
ACSIDEAREA	Side view projected area (sq ft)
ACSKIDFUSE	Height of body center above skid (helos) (ft)
ACSLAT	Seconds of latitude
ACSLONG	Seconds of longitude
ACSPAN	Wingspan (ft)
ACSTATE	Structure. Aircraft state conditions
ACTACCL	Time for acceleration maneuver (sec)
ACTAILAREA	Tail area (side view) (sq ft)
ACTAILDET	Probability of detecting a flare from tail view
ACTCLMB	Time for climb maneuver (sec)
ACTCLMBH	Time for vertical climb (helos) (sec)
ACTCLSTART	Time of start of vertical climb (sec)
ACTLAT	Latitude (deg)
ACTLONG	Longitude (deg)
ACTPITCH	Time for pitch maneuver (sec)
ACTPITCH1	during first part of climb maneuver
ACTPITCH2	during last part of climb maneuver
ACTTURN	Time to perform turn maneuver (sec)
ACTURNAC	Maximum acceleration in turn (gees)
ACTVISOF(NRD)	Time radar ceases being visible
ACTVISON(NRD)	Time radar becomes visible
ACTYP,ACLTYP	Type of aircraft
ACVEL	Velocity (knots)
ACVELG(NGUI)	true interpolated
ACVELO	initial
ACVELP(3)	present and two past values
ACVELF	Velocity (fps)
ACVIS	File. Provides visibility data
ACWINGAREA	Wing area (sq ft)
ACWINGCHORD	Chord length across wing (ft)
ACWRTYP	Type of warning receiver on board
ACYAW	Yaw angle (deg from nose) (deg)
ACYAWG(NGUI)	true interpolated
ACYAWO	initial
ACZALT	Altitude above terrain (ft)
ACZALTO	initial
ACZALTP(3)	present and two past values
AIRCRAFT,AIRCRAFT_LIB	Structure. Aircraft variables
ALLJAM	Flag. Radar jammed in all directions
ALT	Altitude entry for SAM propulsion
ALT1,ALT2	Lower and upper atmospheric transitions
ANTIRAD_MISSILE, ANTIRAD_MISSILE_LIB	Structure. Anti-radiation missile variables
ANTPAT1	Antenna pattern for pencil beams
ANTPATS	Antenna pattern for stacked beams
AREACLUT(2)	Area of pulse doppler clutter patches
ARLBDNO	Identifier for frequency band

ARLIB	File. Anti-radiation missile parameters
ARLNBD	Number of frequency bands available
ARMALT	Altitude of ARM (ft)
ARMBD,ARLBD	Structure. Band variables
ARMBDLFR,ARLBDLFR	Lower frequency limit of band (MHz)
ARMBDUFR,ARLBDUFR	Upper frequency limit of band (MHz)
ARMCL	Class of ARM - preprogrammed, opportunistic
ARMHDG	Heading (deg clockwise from North)
ARMKILLT	Time ARM reaches its target for kill
ARMLAT	Latitude (nmi)
ARMLNCH(NRD)	Flag. ARM aimed at that radar
ARMLOFT,ARLLOFT	Initial loft angle (deg)
ARMLONG	Longitude (nmi)
ARMN	Number of ARMs initially on aircraft
ARMNL	Number of ARMs left on aircraft
ARMOFF	Subroutine. Set parameters to drop ARMs
ARMPAR	Structure. Individual ARMs
ARMPITCH	Pitch angle (deg from horizontal)
ARMPKFACT	Reduction factor for ARM kill probability
ARMKPROB	ARM kill probability
ARMRD	Target radar
ARMRMAX,ARLRMAX	Maximum range capability (nmi)
ARMRMIN,ARLRMIN	Minimum range capability (nmi)
ARMRSLT	Result of attack
ARMSTATE	State of ARM (char)
ARMTDLY,ARLTDLY	Time of flight without guidance
ARMTLNCH	Time of launch (sec)
ARMTOFF	Time target radar goes off and ARM loses guidance
ARMTYP,ARLTYP	Type of ARM
ARMVEL,ARLVEL	Velocity of ARM (fps)
ASACCL	Flag. Aircraft accelerating
ASADVDET	Time a radar first points at an aircraft
ASASSNNO	Temporary tracker assignment number
ASBLANK	Flag. Aircraft blanked from radars
ASBLINKON	Flag. Engaged in blink jamming
ASCARRY	Carrier aircraft for cruise missile
ASCLMB	Flag. Climb in progress
ASDEAD	Flag. Aircraft shot down or outside terrain
ASENG	Number of engagements of aircraft with SAMs
ASFIRDET	Flag. Aircraft has been detected by somebody
ASFLNEXT	Next flare launcher to fire
ASFLPAT	Choice - flare pattern
ASHOMFLG	Flag. Aircraft going home
ASIBLINK	Flag. Capable of blink jamming
ASIDECOY	Flag. Aircraft is a towed decoy
ASIFLPATH	Index of flight path associated with aircraft
ASIHIDE	Aircraft exposed(0), rotor exposed(1), hiding(2)
ASIHIDE0	initial
ASINBEAM(NRD)	Flag. Aircraft scanned by radar this time step
ASINFLD	Flag. Aircraft in field of action

ASIRDET(NRD)	Flag. Launch from that IR site has been detected
ASJMFLG	Flag. Jammer on
ASKFLPATH	Identification no. of associated flight path
ASKFLPATH0	initial
ASLFLARE(2)	Number of flares remaining at each launcher
ASLFLARE0	initial
ASLOAD	Identification for cruise missile while on board carrier
ASMFLARE(2)	Total number of flares fired from each launcher
ASNCRASH	Number of times aircraft has apparently crashed
ASNDECOY	Number of towed decoys on aircraft
ASNFLARE(2)	Number of flares in flight from each launcher
ASPARTNER	Partner for blink jamming
ASPECT	Angle of observation from nose of ac (deg)
ASPTCH	Flag. Aircraft pitching
ASRAKFLG(NRD)	Flag. Acquisition radar from isolated site k has been assigned
ASRAMFLG(NRS)	command site m
ASRTKFLG(NRT)	Flag. Tracker from isolated site k is on
ASRTMFLG(NRT)	command site m
ASSFLG,ASSNFLG	Flag. Used to assign trackers
ASSN	Subroutine. Finds possibility of assignment
ASTOTAL	Number of times aircraft is scanned
ASTRN	Flag. Turn maneuver in progress
ASVCLMB	Flag. Aircraft (helicopter) in vertical climb
ATMOS	Subroutine. Atmospheric variables
ATTN	Attenuation of clutter in doppler filter
A1	Constant, 62089911, in random number generator
BANDFLG	Flag. Radar not in receiver's frequency coverage
BETA	Angle between velocity and line of sight (deg)
BL1	Backlobe level (dB)
BLUMX	File. Flight paths from BLUE MAX (fixed-wing)
C	Velocity of light (.16188 nmi/microsec)
CDENS,CP11,CP12,CP13, CP21,CP22,CP31,CP32, CT11,CT21,CT31,CT32,CVS	Atmospheric constants
CH,CH1,CH2	Dummy characters
CHAMP	File. Flight paths from CHAMP (helicopters)
CLIMB	Subroutine. Climb maneuvers
CLPWR(2)	Power returned from pulse doppler clutter points
CLTSK1	Subroutine. Clutter for pulse doppler seekers
CLTSK2	Subroutine. Clutter for CW seekers
CLUTTER	Subroutine. Clutter for search and track
COMMAND_SITE	Structure. Command site variables
COORD	Coordinate choice rectangular or global on input or output
COR	Correlation parameter
COUNT	Number of maneuvers performed
CROSS	Flag. Radar has scanned across axis
CSALT	Altitude above sea level (ft)
CSCOMALERT(NAC)	Flag. Broadcasting alert on that aircraft

CSDEAD	Flag. Command site not operating
CSDECDLY	Decision delay at command site (sec)
CSDLAT	Degrees of latitude
CSDLONG	Degrees of longitude
CSHLAT	Hemisphere of latitude (N,S)
CSHLONG	Hemisphere of longitude (E,W)
CSIDENT	Identification number of command site
CSLAT	Latitude of command site (nmi)
CSLONG	Longitude of command site (nmi)
CSMLAT	Minutes of latitude
CSMLONG	Minutes of longitude
CSPSMALT(NAC)	Best data on ac altitude at command site
CSPSMFLG(NAC)	Flag. Data to pass
CSPSMLAT(NAC)	Best data on ac latitude at command site
CSPSMLONG(NAC)	longitude
CSSLAT	Seconds of latitude
CSSLONG	Seconds of longitude
CSTLAT	Command site latitude (deg)
CSTLONG	Command site longitude (deg)
CSZALT	Altitude above terrain (ft)
dbuf(20)	Buffer for graphics data
dnitems	Number of items in dbuf
dstring(133)	Buffer string for storage
dtag	Identification for graphics variables
dtype	Variable type for graphics (float=7)
DENS	Atmospheric density (kg/cu.m)
DESTFLG	Flag. SAM should destruct
DETECTIONS	Structure. Detection variables
DETSUM	Structure. Detection summary variables
DIFAZ	Clutter in azimuth difference channel
DIFEL	Clutter in elevation diference channel
DIFFSIG(2)	Difference signals for monopulse receiver
DIFLONG	Longitude from reference
DISPL	File. Graphics output
DLAT	Degrees of latitude
DLATO	initial
DLONG	Degrees of longitude
DLONGO	initial
DLYMULT	Multiplier for up down and decision delays
DR	Degrees to radians (PI/180)
DROPACQ	Subroutine. Drops acquisition
DROPTCK	Subroutine. Drops track
DT	Time step (sec)
DTBRG	Bearing of target, for communication
DTEL	Elevation of target, for communication
DTFLG	Flag. Aircraft detected
DTG	Subdivided time step (sec)
DTNEW	Flag. Aircraft detected as new target
DTPSS	Flag. Aircraft detected on this scan
DTPSSL	Flag. Aircraft detected on last scan
DTRNG	Range of target, for communication (nmi)

DTSISD	Number of detection during this viewing
DTSRDET	Range at which aircraft detected (nmi)
DTSRLOS	Range at which aircraft lost from detection (nmi)
DTSTDET	Time at which aircraft detected (sec)
DTSTLOS	Time at which aircraft lost to detection (sec)
DX,DX1	Aircraft to radar abscissa (nmi)
DXY,DXY1	Aircraft to radar ground range (nmi)
DY,DY1	Aircraft to radar ordinate (nmi)
DZ,DZ1	Aircraft to radar altitude (ft)
ENDFLG	Flag. Endgame in progress
EOFFLG	Flag. End of a file has been reached
EPS	Small constant (.00001)
ERR	Error in servo
EXTINCT(2,2)	Extinction coefficient for optical
FACE	Subroutine. Contributing facets for clutter
FACET(*nfacet*)	Structure. Terrain facets for clutter
FAAZF	Azimuth from radar to facet (deg)
FAELF	Elevation from radar to facet (deg)
FAITXF	Longitude terrain index of facet
FAITYF	Latitude terrain index of facet
FALF	Length of line of sight within facet (nmi)
FARHOD	Diffuse reflection coefficient
FARHOM(2)	Specular reflection coefficient, real and imaginary parts
FASIGF	Reflected signal from facet
FATHETA	Slope of facet along LOS
FATYPEF	Terrain type of facet
FAXF	Latitude of facet (deg, nmi)
FAYF	Longitude of facet (deg, nmi)
FAZF	Altitude of facet above sea level (ft)
FLARE	Structure. Flare flight data
FLALT	Altitude above sea level (ft)
FLALTVEL	Vertical component of velocity (ft/sec)
FLAZ	Azimuth flare to missile (deg)
FLBRITE	Brightness of flare (watts/ster)
FLELEV	Elevation flare to missile (deg)
FLFLTIME	Flight time of flare (sec)
FLKFLARE	Number of flare from that launcher
FLLAT	Latitude (deg, nmi)
FLLATVEL	Northward component of velocity (knots)
FLLNCHD	Flag. Flare launched
FLLNCHTIME	Launch time of flare (sec)
FLLONG	Longitude (deg, nmi)
FLLONGVEL	Eastward component of velocity (knots)
FLRNG	Range flare to missile (nmi)
FLSNR	S/N of flare at missile
FLTGT	IR missile flare is directed against
FLVEL	Magnitude of velocity (knots)
FLZALT	Altitude above terrain (ft)
FLPATH	Structure. Flight path variables
FLYFLARE	Subroutine. Fly flares

FPALT(npoints)	Altitude above sea level of point on path (ft)
FPBANK(npoints)	Path bank angle (+ right wing down) (deg)
FPHDG(npoints)	Path heading (deg)
FPIDENT	Path identifier
FPIHIDE(npoints)	Path hiding parameter
FPJLAST	Index of last point on path
FPLAT(npoints)	Latitude of point on path (deg, nmi)
FPLFLPATH	Number of path points per time step DT
FPLONG(npoints)	Longitude of point on path (deg, nmi)
FPLOS1	Time of first LOS on path for JANUS (sec)
FPLOSN	Time of last LOS on path for JANUS (sec)
FPMFLPATH	Number of guidance time elements(DTG) per flight path time step
FPNPITCH(npoints)	Path nose pitch (deg from horizontal)
FPPITCH(npoints)	Path pitch angle (deg from horizontal)
FPTIME(npoints)	Time of point on path (sec)
FPTIMESTEP	Path time step (sec)
FPVEL(npoints)	Path velocity (knots)
FPYAW(npoints)	Path yaw angle (deg)
FPZALT(npoints)	Altitude of path point above terrain (ft)
FRQ	Frequency to set jammer
FUNC(5)	String for radar functions
GEE	Acceleration of gravity (32.174 ft/sec^2)
GLN	Glint error input to servo
GRPFLG	Flag. Group maneuver
HB	Horizontal beamwidth 1/2 power points (deg)
HF	Choice. Height-finders excluded (N)
HLAT	Hemisphere of latitude (N,S)
HLATO	initial
HLONG	Hemisphere of longitude (E,W)
HLONGO	initial
HNPITCH	Subroutine. Helicopter nose-pitch during climb
i,ii	Index. General
I	Index. Search or track radars
IAC	Index. Aircraft in visibility file
IAVAIL	Flag. Launcher availability recalculated each iteration
IAX,IAY	Terrain indices for first point in FACE
IBX,IBY	Terrain indices for second point in FACE
ICLUT	Flag. Clutter included
ICX,ICY	Terrain indices for specular reflection point
ID	Radar identifier
IFLG	Flag. Clutter is visible
IJANUS	Flag. Probabilities for JANUS calculated
ILLUM_RAD	Structure. Illuminator radar parameters
ILOBS	Flag. Special calculations for low observables
IMSFD	Flag. MSFD numbers in radar data
IMULT	Flag. Multipath included
INBEAM	Subroutine. Aircraft in beam for sector scan
INDX	Index for clutter element along path
INIGHT	Flag. Nighttime operation, no IR or optical

INOUTFLG	Flag. Ac has already reached destination and is being checked for track assign
INTERPOL	Subroutine. Interpolates aircraft position
IOPT	Flag. Optical equipments included
IP	Number time steps per search radar print
IPA	Number time steps per aircraft position print
IPG	Number subdivided time steps per SAM print
IRD	Index. Radar in visibility file
IRDATA	Structure. IR and optical parameters
IRATNFAC(2,5,13,7)	Attenuation of signal in band at range, elev, az
IRBRITE(15,3)	Flare brightness at time and velocity (watts/ster)
IRBRITETIME(15)	Times for brightness look-up table (sec)
IRBRVEL(3)	Velocities for brightness look-up table (knots)
IRCDTIME(4)	Times for flare drag table (sec)
IRCDVEL(2)	Velocities for flare drag table (knots)
IRFLATTN(2,6)	Attenuation of flare signal in band at range
IRFLCDS(4,2)	Flare drag coefficient at time and velocity
IRFLINTVAL	Interval between flare releases (sec)
IRFLMASS(4)	Flare mass at time (kg)
IRFLREL	Flare reliability
IRFLVEL	Flare launch velocity (knots)
IRLIB	File. IR and optical data
IRRANGE(6)	Ranges for radiation look-up table (nmi)
IRSIG	Subroutine. Calculate IR signal
IRSIGNAL(2,13,13)	Radiation in band at az and elev angles (watts/ster)
IRSIGNOM	Aircraft type for radiation data
IRSUNATN(2,6)	Attenuation of sun in band at range
IRSUNSTR(2)	Solar radiation in band (watts/ster)
ISA	Index. Communications modification
ISEEING	Flag. Exceptional seeing conditions
ISNOW	Flag. Terrain all snow covered
ISORAD	Subroutine. Pulse doppler clutter points
ISUMMER	Flag. Summer seeing conditions
ISUN	Flag. Sun visible in sky
ITX	Index. Terrain abscissa
ITY	Index. Terrain ordinate
IX	Random number generated
IXC	Random number for clutter
IWET	Flag. Terrain wet
j,jj	Index. General
J	Index. Aircraft
JAM	Structure. Jammer variables
JAMMER_LIB	Structure. Jammer library variables
JD	Aircraft identifier
JLTYP	Number identifying type of jammer in library
JMBAND,JLBAND	Structure. Variables of individual bands
JMBBDW,JLBBDW	Bandwidth of radiated noise (MHz)
JMBCODE,JLBCODE	Code identifying band (char)
JMBCOUNT	Number of radars in band being jammed
JMBERP,JLBERP	Effective radiated power (watts)

JMBFRQ	Frequency of radar being jammed (MHz)
JMBID	Identification of radar being jammed
JMBLFRQ,JLBLFRQ	Lower frequency limit of band (MHz)
JMBNUM,JLBNUM	Number identifying band
JMBPAR,JLBPAR	Structure. Jammer band variables
JMBPOL,JLBPOL	Polarization loss (dB)
JMBPROD	Factor in radar equation
JMBREL,JLBREL	Reliability of the jammer in that band
JMBS,JLBS	Jammer antenna boresight angle (1st leaf) (deg)
JMBUFRQ,JLBUFRQ	Upper frequency limit of band (MHz)
JMCODE,JLCODE	Code identifying jammer (char)
JMDIP,JLDIP	Antenna dip angle (deg)
JMEFF(NRD)	Jamming effectiveness against radar
JMGAIN,JLGAIN	Gain of repeater
JMLIB	File. Jammer library variables
JMNBAND,JLNBAND	Number of frequency bands in jammer
JMNHB,JLNHB	Jammer antenna horizontal beamwidth (deg)
JMNLEAF,JLNLEAF	Number of leaves in jammer rosette antenna
JMNVB,JLNVB	Jammer antenna vertical beamwidth (deg)
JMOUTBFLG(NRD)	Flag. Radar is not in frequency coverage of jammer
JMPHIN	Angular spacing between leaf centers (deg)
JMRDFLG(NRD)	Flag. That radar is being jammed
JMRDTECH,JLRDTECH	Number identifying techniques jammer can use
JMTACTIC(NRD)	Tactic jammer is using against radar
JREP	Index. Monte Carlo repetition
k,kk	Index. General
K	Index. Sites or general
KILLFLARE	Subroutine. Terminate flares
l,ll	Index. General
L	Index. General
LAT	Latitude for coordinate conversion (deg, nmi)
LATMAX	Maximum latitude of field of action (nmi)
LATMIN	Minimum latitude of field of action (nmi)
LAT0	Earth latitude of center of field of action
LINSGHT	Subroutine. Line of sight range, azimuth, elevation, and if over horizon
LNCHCALC	Subroutine. Time and range for SAM to reach target
LOCATION	Structure. Coordinates, ac, radar, site, command site
LONG	Longitude for coordinate conversion (deg, nmi)
LONGMAX	Maximum longitude of field of action (nmi)
LONGMIN	Minimum longitude of field of action (nmi)
LONG0	Earth longitude of center of field of action
m,mm	Index. General
M	Index. Command sites or general
MASK	Subroutine. Ground clutter masked
MCLUT	File. Ground reflection parameters
MDECOY	Total number of decoys included
MLAT	Minutes of latitude

MLATO	initial
MLONG	Minutes of longitude
MLONGO	initial
MONOPAT	Subroutine. Monopulse antenna pattern
MSKDPR(nmsk)	Depression angles from ridge file
MSKRNG(2,nmsk)	Ranges from ridge file
MTF	Meters to feet (3.2808333)
MULTIP	Subroutine. Calculates multipath
M2	Random number limit value (2147483647)
n,nn	Index. General
ncp	Maximum number of change points permitted (100)
nfacet	Maximum number of facets permitted (200)
nflare	Number of flares per launcher
nmsk	Maximum number of masks along direction (201)
nrdgangle	Number of angles for ridges (37)
nttype	Number of terrain types (9)
NAC	Number of aircraft
NACS	Number of aircraft surviving
NAC1	NAC + 1
NBLUMX	Number of BLUEMAX (fixed-wing) flight paths used
NCHAMP	Number of CHAMP (helicopter) flight paths used
NCP	Maximum number of change points in profile
NDOF	Number of degrees of freedom for aircraft (3,5)
NF	Number of facets
NFACET	Number of facets in FACE subroutine
NFLPATH	Number of flight paths used (total)
NGUI	Number of subdivided time steps per full time step
NORMC	Subroutine. Normally distributed variables for clutter
NORMV	Subroutine. Normally distributed variables
NOSHT	Choice - SAMs do not shoot (Y)
NRD	Number of radars - total
NREP	Number of Monte Carlo Repetitions
NRT	Number of track radars
NSC	Number of command sites
NSI	Number of sites
NTAB	Line number in visibility file
NX	Number of points on terrain abscissa
NY	Number of points on terrain ordinate
OCCUR	Time when detection occurs
OPDET	Subroutine. Optical detection
ORIENT	Subroutine. External to boresight coordinates
OUTPUT	Subroutine. Prints summaries
PAINT	Paint from IRLIB (2-char)
PAST	Past value of correlated error
PD	Probability of detection
PHI	Azimuth entry for LINSGHT and ANTPAT1 (deg)
PHIG	Azimuth to true interpolated position (deg)
PHII	corrupted
PHI1,PHI2	Azimuthal outputs from LINSGHT (deg)

PI	3.14159256
PITCOOR	Subroutine. Coordinates during pitch
PITCH	Subroutine. Pitch maneuver
PMC	Test value for error probability
POINT(2,2)	Calculation points for clutter (nmi)
POWER	Subroutine. Clutter power
PRBTYP	Choice - smooth or cookie-cutter probability of detection, jamming table
PRDBN(NAC)	Structure. Intervals data for JANUS
PRDEND(NRD,10)	End of interaction interval (sec)
PRDINTNO(NRD)	Index for interaction interval
PRDMAX(NRD,10)	Latest end of interaction interval (sec)
PRDMIN(NRD,10)	Earliest start of interaction interval (sec)
PRDNUMB(NRD,10)	Number of kills in interaction interval
PRDPROB1	Probability of kill before first LOS
PRDPROBN	Probability of kill after last LOS
PRDSTART(NRD,10)	Start of interaction interval (sec)
PRES	Present value of correlated error
PRNTMOD	Choice - short radar printout (S)
PROFILE(NCP)	Structure. Maneuver and cutting variables
PROFCHANGE	Description of maneuver (5-char)
PROFTIME	Time of maneuver (sec)
PROFX	First parameter for maneuvers
PROFY	Second parameter for maneuvers
PWRCLT	Total power in clutter
PWR1,PWR2	Jamming power at radar (watts)
P0	Atmospheric constant
RADAR_LIB	Structure. Radar library parameters
RAD	Structure. Parameters for all radars
RCLUT(2)	Range to pulse doppler clutter
RCTDISTRIB	Reaction time distribution (N,U)
RDACOH(NAC)	Flag. Aircraft over horizon to radar
RDALT	Altitude (ft)
RDALERT	Flag. Optical system in alert condition
RDALERTED(NAC)	Flag. Optical system alerted to that aircraft
RDAMAX(NAC),RSAMAX RTAMAX,RIAMAX	Maximum range against that aircraft (nmi)
RDAPAR,RSAPAR,RTAPAR, RIAPAR	Structure. Additional radar parameters
RDARMSOFF	Flag. ARMs aimed at radar should destruct
RDATSITE	Site at which radar is located
RDBL,RDLBL,RSBL,RTBL, RIBL	Backlobe level (dB)
RDCLAS	Class - search, HF, acquisition, track, IR, gun
RDCORNERANGLE(4)	Angle - radar to four corners of terrain (deg)
RDDLAT	Degrees of latitude
RDDLONG	Degrees of longitude
RDERP,RDLERP,RSERP, RTERP,RIERP	Effective radiated power (watts)
RDFRQ,RSFRQ,RTFRQ,RIFRQ	Frequency (MHz)
RDFUNC	Functions performed by radar (4-char)

RDGAN,RDLGAN,RSGAN, RTGAN,RIGAN	Antenna gain (dB)
RDHLAT	Hemisphere of latitude (N,S)
RDHLONG	Hemisphere of longitude (E,W)
RDIDENT	Radar identifier
RDIFUNC	Number of functions performed by radar
RDJMTECH,RDLJTECH, RSJMTECH,RTJMTECH, RIJMTECH	Most effective jamming technique
RDJPROD,RSJPROD,RTJPROD, RIJPROD	Factor in range equation
RDLAT	Latitude (nmi)
RDLIB	File. Radar library parameters
RDLFRQ,RDLLFRQ,RSLFRQ, RTLFRQ,RILFRQ	Lower frequency limit (MHz)
RDLOSS,RDLLOSS,RSLOSS, RILOSS	Loss in cables and connectors (dB)
RDLNAME	Name of radar type in library
RDMAX,RDLMAX,RSMAX, RTMAX	Maximum range against 1 sqm target (nmi)
RDMJS,RSMJS,RTMJS,RIMJS	Thermal noise power (watts)
RDMLAT	Minutes of latitude
RDMLONG	Minutes of longitude
RDMSFDNO	Radar identification number in MSFD threat laydown
RDNBW,RDLNBW,RSNBW, RTNBW,RINBW	Receiver bandwidth (MHz)
RDNHB,RDLNHB,RSNHB RTNHB,RINHB	Horizontal beamwidth 1/2 power points (deg)
RDNMASK(nrdgangle)	Number of masks in given direction
RDNVB,RDLNVB,RSNVB RTNVB,RINVB	Vertical beamwidth 1/2 power points (deg)
RDPAR,RDLPAR,RSPAR RTPAR,RIPAR	Structure. Radar parameters
RDPRF,RDLPRF,RSPRF RTPRF,RIPRF	Pulse repetition frequency (Hz)
RDPW,RDLPW,RSPW,RTPW, RIPW	Pulse width (microsec)
RDRDG	File. Ridges produced by RJARS calculation
RDRIDGDIST(nrdgangle, 2,NMASK)	Distance to each line of sight disappearance(1) and reappearance(0) in given direction (nmi)
RDRIDGDPR(nrdgangle, NMASK)	Angle of depression to each mask (deg)
RDRIDGFLG	Flag. Radar in ridge list
RDSCANDIR	Direction of scan
RDSCANMAX	Maximum angle of scan coverage (deg)
RDSCANMIN	Minimum angle of scan coverage (deg)
RDSCANWIDTH	Half-angle of scan coverage (deg)
RDSCN,RDLSCN,RSSCN, RTSCN,RISCN	Scan or settling period (sec)
RDSECTMAX	Maximum angle of sector coverage (deg)

RDSECTMEAN	Mean angle of sector coverage (deg)
RDSECTMIN	Minimum angle of sector coverage (deg)
RDSECTWIDTH(2)	Half-width of sector coverage, uncued or cued (deg)
RDSIGMAO(nttype), RSSIGMAO,RTSIGMAO,	Scattering coefficient per unit area of terrain type at radar frequency
RDSLAT	Seconds of latitude
RDSLONG	Seconds of longitude
RDSMTYP,RDLSMTYP, RTSMTYP	SAM type associated with radar
RDSPROD,RSSPROD, RTSPROD,RISPROD	Factor in radar equation
RDTLAT	Latitude (deg)
RDTLONG	Longitude (deg)
RDTMAX,RDLTMAX,RSTMAX, RTTMAX	Maximum time target held by jammed radar
RDTYP,RDLTYP	Type of radar
RDUFRQ,RDLUFRQ,RSUFRQ, RTUFRQ,RIUFRQ	Upper frequency limit (MHz)
RDVUL(5),RDLVUL(5), RSVUL(5),RTVUL(5), RIVUL(5)	Vulnerability to ARM type
RDWPROD,RSWPROD,RTWPROD, RIWPROD	Factor in warning receiver equation
RDZALT	Height above terrain (ft)
RECT	Subroutine. Global to rectangular
REORIENT	Subroutine. Boresight to external coordinates
RESPNS	Subroutine. Monopulse antenna response
RH,RHL	Radar horizon (nmi)
RIDGE	File. Ridges input
RIDGES	Subroutine. Calculates ridges
RIDGSPAC	Angular spacing between ridges (deg)
RIDGTEST	Subroutine. Tests if ground visible
RIDOPF(NAC)	Doppler frequency of target (Hz)
RNG	Range output from LINSGHT
RNGG	Interpolated true range (nmi)
RNGI	corrupted
RSACNO,RTACNO	Aircraft assigned to radar
RSAFLG,RTFLG	Flag. Radar assigned
RSANT	Direction antenna is pointing (deg)
RSANTO	initial
RSANTCH	Change in antenna direction per time step
RSANTNREF	Antenna direction incl < 0 and > 360
RSARMN,RTARMN	Number of ARMs aimed at radar
RSAZERR	Azimuth error (deg)
RSCLUTAZER,RTCLUTAZERR	Azimuth error produced by clutter (deg)
RSCLUTELER,RTCLUTELER	Elevation error produced by clutter (deg)
RSCLUTSIG,RTCLUTSIG	Signal produced by clutter (dB)
RSDEAD,RTDEAD	Flag. Radar has been killed
RSDET	Heading marker
RSELERR	Elevation error (deg)
RSFSB	Far sidelobe inner limit of radar (deg)

RSHITS(NAC)	Number of successful detections
RSIFOV(NAC)	Flag. Aircraft in field of view of radar
RSISF(NAC)	Index - number times in field of view
RSLOOKS(NAC)	Number of times radar scans over aircraft
RSMDS	Receiver processing gain (dB)
RSMESSPROB	Probability of receiving alerting message
RSMULTELER,RTMULTELER	Error in elevation produced by multipath (deg)
RSNO	Search radar number for given radar number
RSNPLS	Number of pulses reflected from target
RSNSTACK, RDLNSTACK	Number of stacked beams in antenna pattern
RSOBSTIME(NAC)	Duration of observation by optical system (sec)
RSPOL,RTPOL	Polarization of signal (H,V)
RSPOSERR,RTPOSERR	Position error (ft)
RSREFOV(NAC,10)	Range at exit from field of view (nmi)
RSRIFOV(NAC,10)	Range at entry into field of view (nmi)
RSRNGERR	Range error (ft)
RSSCVD,RTSCVD	Subclutter visibility (dB)
RSSTATE,RTSTATE	Structure. State of radar
RSTDROP(NAC)	Time to drop detection hold (sec)
RSTEFOV(NAC,10)	Time to exit field of view (sec)
RSTHRESH	Threshold level (watts)
RSTIFOV(NAC,10)	Time of entry into field of view (sec)
RSTON,RTTON,RITON	Time radar comes on in indicated state
RSTRK	Tracker associated with acq radar
RSTSUM(NAC)	Number of successive detections
RTACQ	Acq radar associated with tracker
RTACQOFF(NAC,10)	Time acq radar goes off (sec) for output
RTACQON(NAC,10)	Time acq radar comes on (sec) for output
RTALTI(NGUI)	Interpolated corrupted target altitude (ft)
RTALTP(3)	Corrupted altitude now and last two (ft)
RTASSN(NAC,10)	Time tracker assigned (sec) for output
RTAZ,RIAZ	Azimuth of tracker (deg)
RTEL,RIEL	Elevation of tracker (deg)
RTGLNAZCOR	Correlated azimuth
RTGLNELCOR	Correlated elevation
RTHDGP(3)	Corrupted heading now and last two (deg)
RTITK(NAC)	Index. Number times tracker assigned to aircraft
RTJAMFLG	Flag. Tracker is being jammed, SAM launched
RTJMCNT	Number time steps tracker has been jammed, SAM not launched
RTJS	Signal/noise ratio at intercept
RTLATI(NGUI)	Interpolated corrupted latitude (deg)
RTLATP(3)	Corrupted latitude now and last two (deg)
RTLONGI(NGUI)	Interpolated corrupted longitude (deg)
RTLONGP(3)	Corrupted longitude now and last two (deg)
RTMHSB	Maximum azimuthal servo bandwidth (Hz)
RTMVSB	Maximum elevation servo bandwidth (Hz)
RTOFF(NAC,10)	Time tracker goes off (sec) for output
RTOUTPUT	Structure. Tracker summary variables
RTPAST	Structure. Tracker past variables
RTPHIACC	Azimuthal acceleration (deg/sec sq)

RTPHIP(3)	Azimuth past values (deg)
RTPITCHP(3)	Corrupted pitch now and last two (deg)
RTREASSN	Flag. Tracker reassigned
RTRNGACC	Range acceleration (nmi/sec^2)
RTRNGP(3)	Past values of range (nmi)
RTSRB(3)	Servo bandwidth, 3 channels (Hz)
RTTHAZCOR	Correlated azimuth error from noise (deg)
RTTHELCOR	Correlated elevation error from noise (deg)
RTTHETAACC	Elevation acceleration (deg/sec^2)
RTTHETAP(3)	Past elevation (deg)
RTTHRNGCOR	Correlated range error from noise (nmi)
RTTIMNRA	Time tracking - no range, angle (sec)
RTTIMNRNA	no range, no angle (sec)
RTTIMRA	range, angle (sec)
RTTIMRNA	range, no angle (sec)
RTTON(NAC,10)	Time tracker comes on (sec) for output
RTVELP(3)	Corrupted velocity now and last two (knots)
R1,R2	Range outputs from LINSGHT
R90,R270	Constants used in RECT
srchno	Maximum number of times in view (10)
str(4)	Radar type for graphics
SAMACC	Subroutine. SAM acceleration
SAMLIB	Structure. SAM library variables
SCALEH	Atmospheric scale height(34174 ft)
SCENA	File. Scenario for simulation
SDL	Sidelobe attenuation level (dB)
SECTASS	Subroutine. Sector assignment
SECTSCAN	Subroutine. Sector scanning
SEED	Starting value for random number generator
SEEKER	Structure. Seeker parameters
SEKRPAT	Subroutine. Seeker antenna pattern
SIGMA	Radar cross section
SIGMA1	Subroutine. Calculate radar cross section
SIGMA2	Radar cross section output from SIGMA1
SIGNAL	Reflected power received from target
SIMPLE_SAM	Structure. SAM variables
SITE	Structure. Site variables
SIALERT(NAC)	Flag. Site alerted to that aircraft
SIALT	Altitude above sea level (ft)
SICOMFLG	Flag. Communications operating
SIDEAD	Flag. Site not operating
SIDLAT	Degrees of latitude
SIDLONG	Degrees of longitude
SIDNDLY	Delay on downward communication link (sec)
SIHLAT	Hemisphere of latitude (N,S)
SIHLONG	Hemisphere of longitude (E,W)
SIIDENT	Site identification number
SILAT	Site latitude (nmi)
SILONG	Site longitude (nmi)
SIMESSPROB	Probability of receiving message
SIMLAT	Minutes of latitude

SIMLONG	Minutes of longitude
SIMOBILE	Flag. Site capable of motion
SIMOVING	Flag. Site actually moving
SINRSSI	Number of search radars at site
SINRTSI	Number of track radars at site
SINRTSIA	Number of assigned track radars at site
SIPSALT(NAC)	Best data on ac altitude at isolated site
SIPSFLG(NAC)	Flag. Data to pass
SIPSLAT(NAC)	Best data on ac latitude at isolated site
SIPSLONG(NAC)	longitude
SIRPT	Command site to which site reports
SIRSSIID,SIRTSSID	Sequence number matching identification number in list of radars at site
SISLAT	Seconds of latitude
SISLONG	Seconds of longitude
SITLAT	Latitude (deg)
SITLONG	Longitude (deg)
SIUPDLY	Delay on upward communication link (sec)
SIZALT	Altitude above terrain (ft)
SIZE	Character for number of change points (S,M,L)
SJ1,SJ2	Signal to noise ratio
SKAIMERR	RMS aiming error for guns (deg)
SKAMBIG	Ambiguous range of pulse doppler (nmi)
SKATTNMX	Maximum attenuation of doppler filter (dB)
SKAUTON	Flag. 1 if autonomous
SKAZ(2)	Azimuth present and past (deg)
SKAZCOM(3)	Azimuth command now and last two (deg)
SKAZSK(3)	Azimuth from boresight now and last two (deg)
SKBL	Seeker backlobe level (dB)
SKCLLEN	Length of clutter element (nmi)
SKCLUTSIG	Signal from clutter (dB)
SKDELTAF	Doppler bank filter width (Hz)
SKDFILTW	MTI filter width (Hz)
SKDOPFTM	Doppler frequency target to missile (Hz)
SKELCOM(3)	Elevation command now and last two (deg)
SKELEV(2)	Elevation present and past (deg)
SKELSK(3)	Elevation from boresight now and last two (deg)
SKFLRJCT(2,NFLARE)	Rejection state of flare from launcher 0 or 1
SKFLRJCTPROB	Probability of rejecting a flare
SKFLRJCTTIME	Time for decision to reject flare (sec)
SKFOV	IR seeker field of view (deg)
SKFOVO	Optical field of view (deg)
SKINTFAC(2)	Factor in blink jamming equation
SKINTERF(2)	Power in blink jamming (watts)
SKIRTYP	Short or long wavelength band
SKJS	Jammer/signal ratio
SKKSERV	Locke's K parameter
SKLOCK	Full, partial, or no lock
SKLRATL	Minimum S/N for seeker motion (dB)
SKMAGNIFY	Magnification of optical system
SKNHB	Seeker horizontal beamwidth (deg)

SKNOISE	Noise (incl jamming) at receiver (watts)
SKNSERV	Locke's N parameter
SKNVB	Seeker vertical beamwidth (deg)
SKRATLMX	Maximum rate of seeker motion (deg/sec)
SKROLOFF	Rate of attenuation dropoff in filter bank
SKSCON(2)	Monopulse receiver constants
SKSERVCON(5)	Servo response constants
SKSIGFAC	Factor in signal equation
SKSIGNAL	Signal at receiver (watts)
SKSIGNOI	Signal/noise ratio
SKTDWELL	Time optical system dwells on target (sec)
SKTHRESH	Threshold signal/noise ratio
SKTYPE	Pulse doppler or CW
SKURATL	S/N required to maximum seeker motion (dB)
SKWSERV	Locke's W parameter
SKXNEI	IR noise equivalent input (watts)
SKYGROUND(2,2)	Sky/ground ratio for optical
SLAT	Seconds of latitude
SLATO	initial
SLONG	Seconds of longitude
SLONGO	initial
SLTYP	Type of SAM
SMALT	Altitude (ft)
SMAREF,SLAREF	Reference area (sqm)
SMAVAIL	Flag. SAM is operating
SMAZ	Azimuth - launcher to SAM (deg)
SMAZPM(2)	Azimuth - SAM to partner now and last (deg)
SMAZPRO(2)	Azimuth - SAM to target projected position now and DTG ago (deg)
SMAZTM(4)	Azimuth - SAM to target now, two previous times, and true (deg)
SMCD1(4,3),SLCD1(4,3), SMCD2(4,3),SLCD2(4,3)	Constants in coefficient of drag expression
SMCEP,SLCEP	Dispersion of SAM guidance system (ft)
SMCOUNT	Number of SAMs fired from launcher
SMELEV	Elevation - launcher to SAM (deg)
SMELEVPM(2)	Elevation - SAM to partner now and last (deg)
SMELEVPRO(2)	Elevation - SAM to target projected position now and DTG ago (deg)
SMELEVTM(4)	Elevation - SAM to target now, two past, and true (deg)
SMFUZRNG	Fuze range (ft)
SMFUZTYPE	Fuze type. 0 contact, 1 proximity
SMGUICL,SLGUICL	Guidance class (A,C,S,I)
SMGUIGAN,SLGUIGAN	Gain in guidance loop
SMHDG(2)	Heading of SAM - now and DTG ago (deg)
SMHMAX,SLHMAX	Maximum operating altitude (ft)
SMHMIN,SLHMIN	Minimum operating altitude (ft)
SMINT(20)	Outcome of interception for output
SMINTR(20)	Range at intercept (nmi) for output
SMINTT(20)	Time of intercept (sec) for output

SMJAMSUSC(2,20), SLJAMSUSC(2,20)	Susceptibility to that jammer in each table
SMJAMTYP(2,20), SLJAMTYP(2,20)	Type of jammer for effectiveness tables
SMJSCRIT(20), SLJSCRIT(20)	Critical S/N for that jammer
SMKPROB(20)	Kill probability at intercept for output
SMKRMAX,SLKRMAX	Radius. Kill prob 0 outside (ft)
SMKRMIN,SLKRMIN	Radius. Kill prob = warhead reliability inside it (ft)
SMLAGCON(4)	Constants in guidance lag equation
SMLAT	Latitude (nmi)
SMLIB	File. SAM variables
SMLNCFLG	Flag. SAM preparing for launch
SMLNCHD	Flag. SAM launched
SMLOFT,SLLOFT	Initial loft angle (deg)
SMLOFTRT	Rate of decrease of pitch in boost (deg/sec)
SMLONG	Longitude (nmi)
SMLREL,SLLREL	Launch reliability
SMLSUC(20)	Launch outcome (SUCCESS,FAILURE) for output
SMLTIME(20)	Time of launch (sec) for output
SMMACHT(4,3), SLMACHT(4,3)	Transition value of Mach number
SMMASS	Present mass of SAM (kg)
SMMASSRT(4),SLMASSRT(4)	Mass flow rate of each stage (kg/sec)
SMMASSO(4),SLMASSO(4)	Initial mass of each stage (kg)
SMMISSD(20)	Miss distance at intercept (ft) for output
SMMNVEL	Mean velocity (fps) Used in LNCHCALC
SMMOBLTY,SLMOBLTY	Probability SAM is mobile
SMNRT,SLNRT	Number SAMs initially available at launcher
SMNRTA	Number SAMs presently available at launcher
SMNSTO,SLNSTO	Number SAMs initially in storage at launcher
SMNSTOA	Number SAMs presently in storage at launcher
SMNSUSC(2),SLNSUSC(2)	Number of jammers with data in tables
SMOUTPUT	Structure. Output summary variables
SMPITCH(2)	Pitch angle of velocity. Now and DTG ago (deg)
SMPITCOM(2)	Pitch command now and past
SMPRNT	Choice - print SAM data each IPG'th subdivided time step (Y)
SMPROPUL,SLPROPUL	Structure. Propulsion variables
SMRBECO,SLRBECO	Range at booster cutoff (nmi)
SMRINT	Range at intercept (nmi)
SMRMAX,SLRMAX	Maximum operating range (nmi)
SMRMIN,SLRMIN	Minimum operating range (nmi)
SMRNG	Range launcher to SAM (nmi)
SMRNGPM(2)	Range to partner now and previous (nmi)
SMRNGTM(3)	Range SAM to target - past, present, true (nmi)
SMRSECO,SLRSECO	Range at sustainer cutoff (nmi)
SMSTAGE	Present stage of SAM
SMSTATE	Character giving SAM state (A,F,P,B,I,S,C,R,E)
SMSYSREL,SLSYSREL	Overall system field reliability

SMTBECO,SLTBECO	Time of booster cutoff (sec)
SMTGT(20)	SAM target aircraft for output
SMTHRUST(4),SLTHRUST(4)	Thrust of each stage (Newtons)
SMTIMCON,SLTIMCON	Time constant for guidance lag
SMTINT	Time of intercept (sec)
SMTINTER,SLTINTER	Duration of interstage interval (sec)
SMTLNCH	Time of launch (sec)
SMTMAX,SLTMAX	Maximum flight time till self-destruct (sec)
SMTRCT,SLTRCT	Reaction time - first launch against ac (sec)
SMTRCTMAX,SLTRCTMAX	Maximum reaction time for distribution (sec)
SMTRCTMEAN	Mean reaction time for distribution (sec)
SMTRCTMIN,SLTRCTMIN	Minimum reaction time for distribution (sec)
SMTRCTO	Initial reaction time (sec)
SMTRCTSD	Standard deviation for distribution (sec)
SMTREF,SLTREF	Reaction time - refiring against ac (sec)
SMTREFMAX,SLTREFMAX	Maximum refire time for distribution (sec)
SMTREFMEAN	Mean refire time for distribution (sec)
SMTREFMIN,SLTREFMIN	Minimum refire time for distribution (sec)
SMTREFO	Initial refire time (sec)
SMTREFSD	Standard deviation for distribution (sec)
SMTRLD,SLTRLD	Reload time - storage to launcher (sec)
SMTRST	Restart time after reloading (sec)
SMTSECO,SLTSECO	Time of sustainer cutoff (sec)
SMVEL	Velocity (fps)
SMWREL,SLWREL	Warhead reliability
SMYAWCOM(2)	Yaw command now and past
SMZALT	Altitude above terrain (ft)
SPHER	Subroutine. Rectangular to global
SRBMOD	Choice - adaptive or fixed servo bandwidth (A,F)
SRCH_RAD	Structure. Search radar parameters
SRVERR	Subroutine. Uncorrelated errors
SRVERRCOR	Subroutine. Correlated errors
STORE	Structure. Storage for flight paths
STIDENT	Stored flight path identifier
STLENGTH	Stored flight path number of points
STRX	Global longitude variable in ZTERN
STRY	Global latitude variable in ZTERN
SUMCLT	Clutter in monopulse sum channel
SUMMARY	Structure. Search radar summary variables
SUMSIG	Sum channel output for monopulse receiver
SYNEX,SYNEY	Directions for clutter path
toff	Time of visibility end in file (sec)
ton	Time of visibility start in file (sec)
trans	Type cast from character to float
trkno	Maximum number of times in view (10)
T	Simulation time (sec)
TC	Counter for maneuvers
TCKERR(4)	Tracker error - range (ft), azimuth and elevation (deg), position (ft)
TDAY	Time of day (hr)
TEALT	Altitude of terrain point above sea level

TEMP,TEMP1	Dummy calculation variables
TERMINATE	Subroutine. Eliminate aircraft if outside terrain
TERRA	File. Terrain data
TERRAIN(2*NX+3,2*NY+3)	Structure. Terrain data
TETYP	Terrain type
TFIN	Final time of simulation (sec)
THETA	Input elevation for LINSGHT and ANTPAT1
THETAG	Interpolated elevation to true position (deg)
THETAI	corrupted
THETA1,THETA2	Elevation outputs from LINSGHT
TL	Time of SAM launch (sec)
TOFF	Time visibility ceases - from file (sec)
TON	Time visibility begins - from file (sec)
TRACKON	Subroutine. Turn tracker on
TRK_RAD	Structure. Track radar variables
TRALT	Altitude of terrain point above sea level (ft)
TRDR	Diagonal terrain step (nmi or deg)
TRDX	Terrain abscissa step (nmi or deg)
TRDY	Terrain ordinate step (nmi or deg)
TRINDEX	Subroutine. Finds terrain indices
TRLAT	Latitude of terrain point (deg,nmi)
TRLGMAX	Maximum longitude of terrain point (deg, nmi)
TRLGMIN	Minimum longitude of terrain point (deg, nmi)
TRLONG	Longitude of terrain point (deg, nmi)
TRLTMAX	Maximum latitude of terrain point (deg, nmi)
TRLTMIN	Minimum latitude of terrain point (deg, nmi)
TRPAR(nttype)	Structure. Terrain parameters
TRPA	Parameter A in terrain radar cross-section
TRPB	Parameter B in terrain radar cross-section
TRPC	Parameter C in terrain radar cross-section
TRPCODE	Terrain type description
TRPD	Parameter D in terrain radar cross-section
TRPEPSI(2)	Imaginary part of dielectric coeff (dry, wet)
TRPEPSR(2)	Real part of dielectric coefficient (dry, wet)
TRPFRESCO(2,2)	Real and imaginary parts of Fresnel coefficient (dry, wet)
TRPREFL	Visible reflectivity
TRPROUGH	RMS roughness parameter (cm)
TRPTYPE	Index. Terrain type
TRTYP	Type of terrain element
TRX	Terrain longitude used in TRINDEX
TRY	Terrain latitude used in TRINDEX
TSAM	Subdivided time for SAM calculations (sec)
TTC	Time counter
TURN	Subroutine. Turn calculation
TYPE(5)	String for radar types
TO	Atmospheric constant
UNIV	Subroutine. Uniformly distributed variables
UNIVC	Subroutine. Uniform distribution for clutter
UPDAC	Subroutine. Aircraft
UPDAR	Subroutine. Anti-radiation missiles

UPDCK	Subroutine. System clock
UPDLY(NSI)	Uplink delay from site to command site (sec)
UPDMC	Subroutine. Monte Carlo
UPDRS	Subroutine. Search radars
UPDRT	Subroutine. Track radars
UPDSI	Subroutine. Sites
UPDSM	Subroutine. Surface-to-air missiles
UPDTR	Subroutine. Terrain
UPDWR	Subroutine. Warning receivers
var0,var1,var2,var3, var4,var5,var6,var7, var8,var9,var10,var11,var12	Dummy calculation variables
VAR1,VAR2,VAR3,VAR4, VAR5,VAR6,VAR7,VAR8, VAR9,VAR10,VAR11,VAR12	Dummy calculation variables
VB	Vertical beamwidth input for ANTPAT1 (deg)
VEL	Velocity input for SAMACC (fps)
VELAPP	Cosine of angle between ac velocity and line of sight
VIS	Choice - terrain mode
VISDATA(NAC,NRD)	Structure. Visibility data
VISKEY	Key for visibility data sequence on each pair
VISTOFF(20)	Time aircraft ceases to be visible to radar
VISTON(20)	Time aircraft becomes visible to radar
VSND	Velocity of sound (fps)
WARNING_RECEIVER	Structure. Warning receiver variables
WARNING_RECEIVER_LIB	Structure. Warning receiver library
WLTYP	Type of warning receiver
WRBAND,WLBAND(20)	Structure. Frequency band variables
WRBCODE,WLBCODE	Band identifying code (char)
WRBCOUNT	Number of received signals in band
WRBGAN,WLBGAN	Antenna gain in band (dB)
WRBID(20)	Identification of radar in list in band
WRBLFRQ,WLBLFRQ	Lower frequency limit of band (MHz)
WRBNUM,WLBNUM	Number of band
WRBPAR,WLBPAR	Structure. Band parameters
WRBS,WLBS	Angle from nose of first rosette leaf (deg)
WRBSENS,WLBSENS	Sensitivity of band receiver (dB/milliwatt)
WRBUFRQ,WLBUFRQ	Upper frequency limit of band (MHz)
WRDIP,WLDIP	Elevation boresight of antenna (deg, + up)
WRJMOFF	Subroutine. Turns jammer off
WRJMON	Subroutine. Turns jammer on
WRLIB	File. Warning receiver library
WRNBAND,WRLBAND	Number of frequency bands in a receiver
WRNHB,WLNHB	Horizontal beamwidth of rosette leaf (deg)
WRNLEAF,WLNLEAF	Number of leaves in rosette
WRNVB,WLNVB	Vertical beamwidth of rosette leaf (deg)
WRPHIN	Angle between azimuth leaf boresights (deg)
WRRDFLG(NRD)	Flag. Radar has been detected and catalogued
WRRECOFF	Subroutine. Turns warning receiver off
X	Input variable for subroutines

XA	Elliptical angle for ANTPAT1
XCLUT(2)	Longitudes for pulse doppler clutter points (nmi)
XG	Longitude for clutter point (nmi)
XL	Longitude reference point for multipath (nmi)
XMOM(2)	Longitude moment of clutter
XR	Variable for random numbers (double precision)
XRC	Variable for clutter random number
X1,X2,XD	Dummy longitudes for clutter (nmi)
Y	Input variable for subroutines
YCLUT(2)	Latitudes for pulse doppler clutter points (nmi)
YFL	Uniformly distributed variable from UNIV
YG	Latitude for clutter point (nmi)
YL	Latitude reference point for multipath
YMOM(2)	Latitude moment for clutter
YNC(2,2)	Slopes for clutter lines
Y1,Y2,YD	Dummy latitudes for clutter (nmi)
Z	Output variable from ZTERN
ZL	Altitude of multipath reference point (ft)
ZTERN	Subroutine. Calculates height of terrain
ZTRX	Longitudinal variable in ZTERN
ZTRY	Latitudinal variable in ZTERN

Appendix

FLOW CHARTS

Following a suggestion of the reviewer, a set of flow charts is included in this appendix to show the operation of the program in graphic form. The charts are arranged so that a block on the chart usually represents a block of code that performs the operation described in the words accompanying the block. The first chart, Fig. A.1, gives the overall structure of RJARS, and the remaining charts, Figs. A.2 - A.13, show in greater detail the actions in each functional block of Fig. A.1. In the author's view, the charts can best be used by referring to them while reading the text.

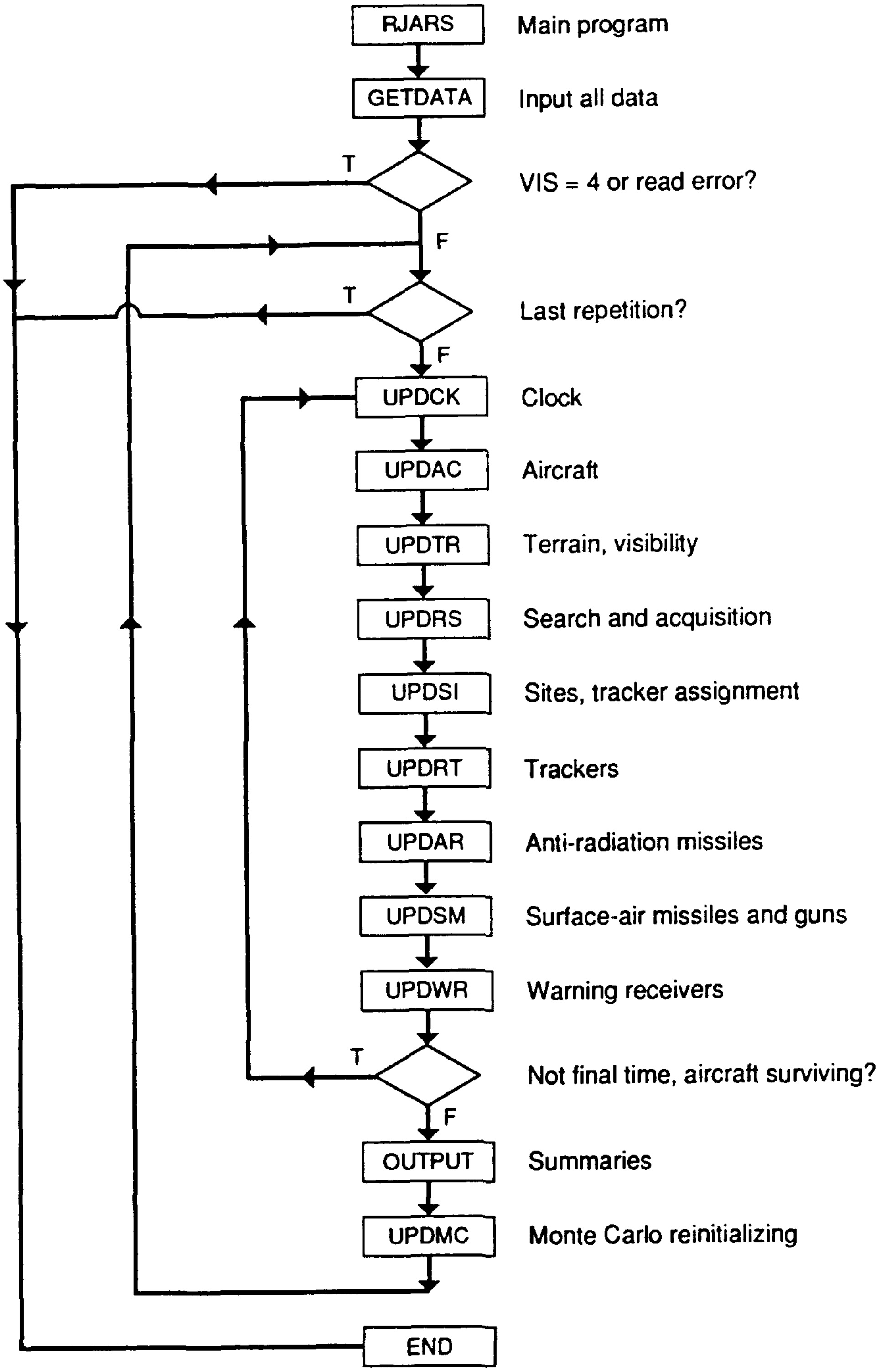


Fig. A.1—RJARS flow chart

Sequence and input file		
1.	Initialization constants	SCENA
2.	Terrain data	TERRA
3.	Aircraft scenario	SCENA
4.	Aircraft physical data	ACLIB
5.	Aircraft radar cross-section	ACRCS
6.	Anti-radiation missile data	ARLIB
7.	Jammer equipment data	JMLIB
8.	Warning receiver equipment data	WRLIB
9.	Flight profile	SCENA
10.	Flight path data	BLUMX, CHAMP
11.	Infrared equipment and radiation data	IRLIB
12.	Radar scenario	SCENA
13a.	Radar ridge data	RIDGE
13b.	Radar ridge calculation	SCENA, TERRA
14.	Radar equipment data	RDLIB
15.	Site and command site data	SCENA
16.	Surface-air missile data	SMLIB

Fig. A.2—GETDATA sequence chart

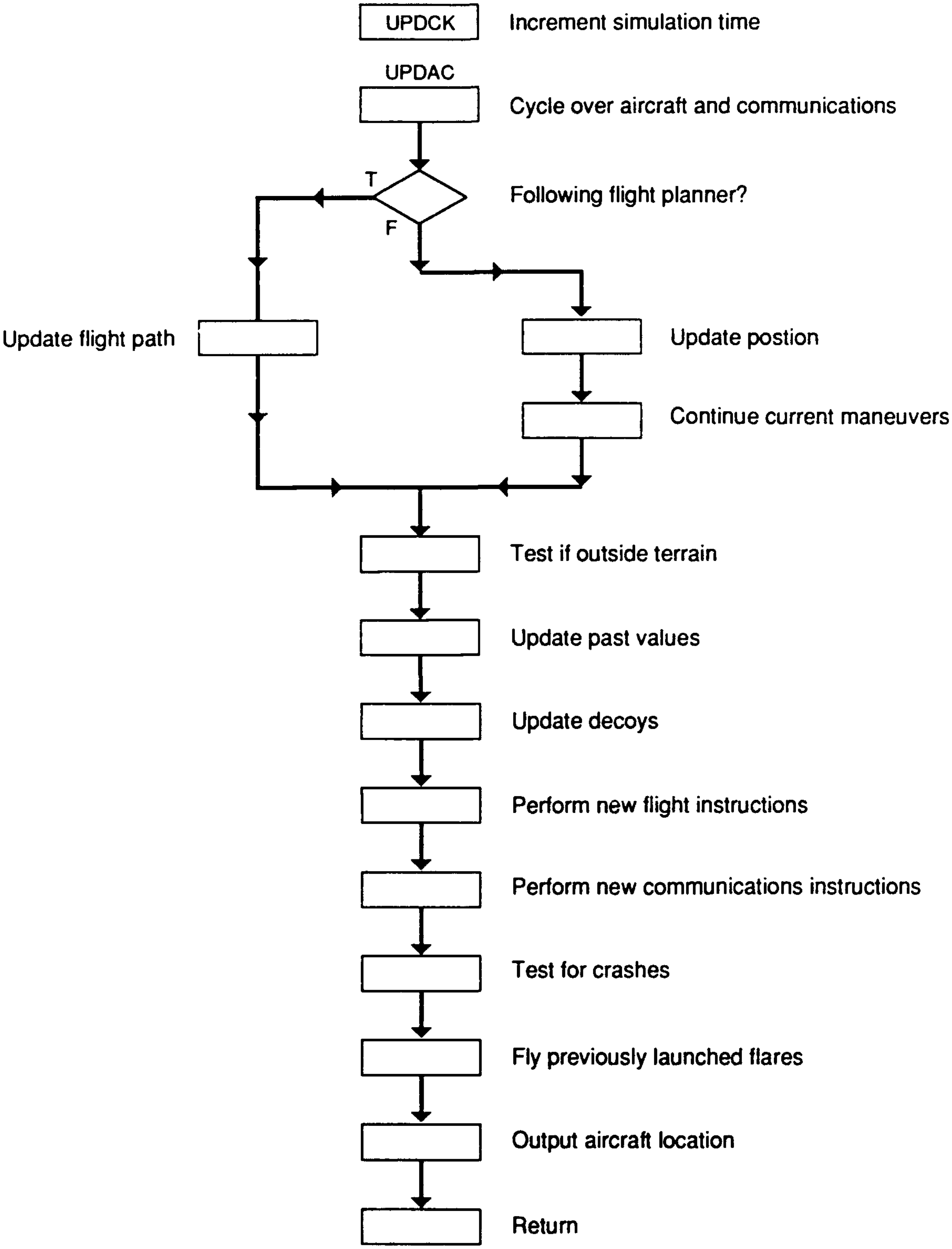


Fig. A.3—Update clock (UPDCK) and update aircraft (UPDAC) flow chart

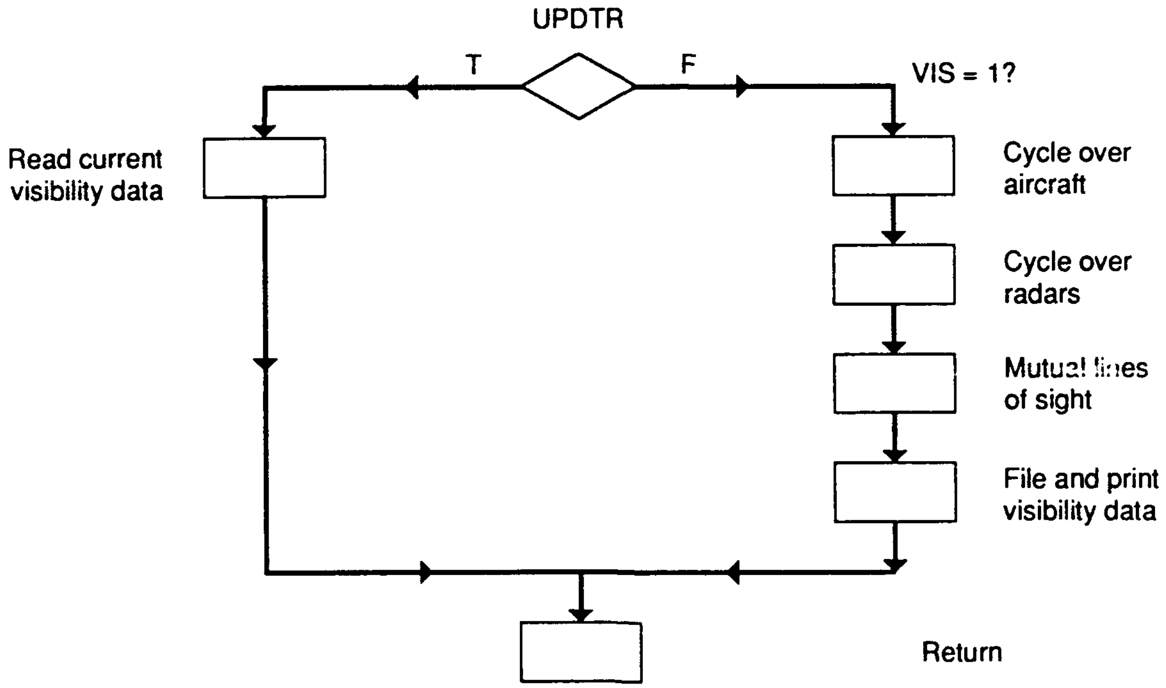


Fig. A.4—Update terrain (UPDTR) flow chart

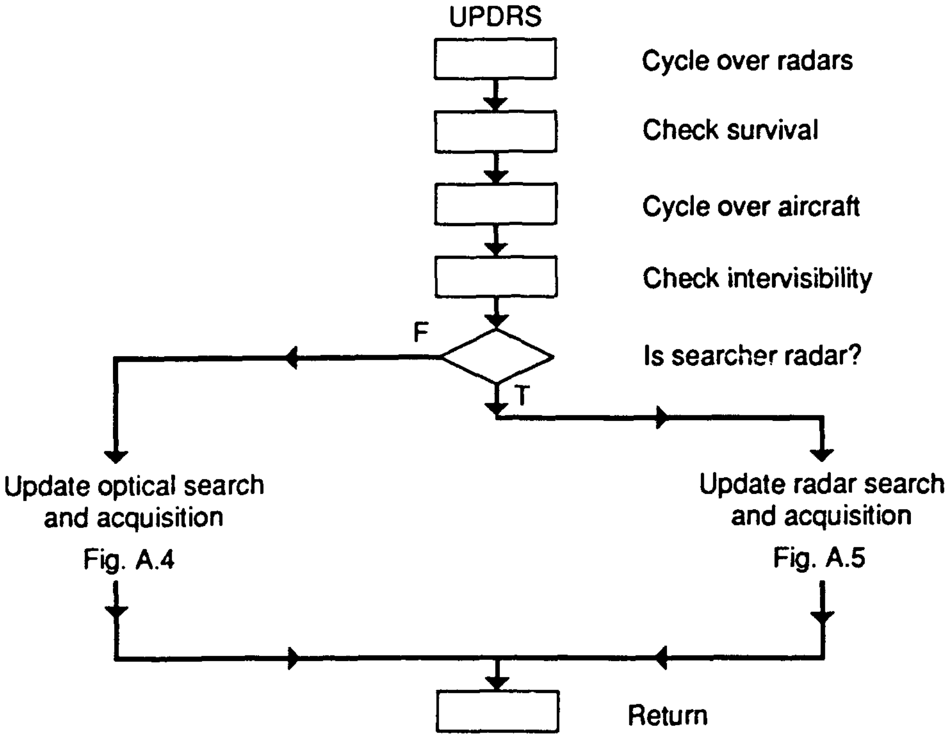


Fig. A.5—Update searchers (UPDRS) flow chart

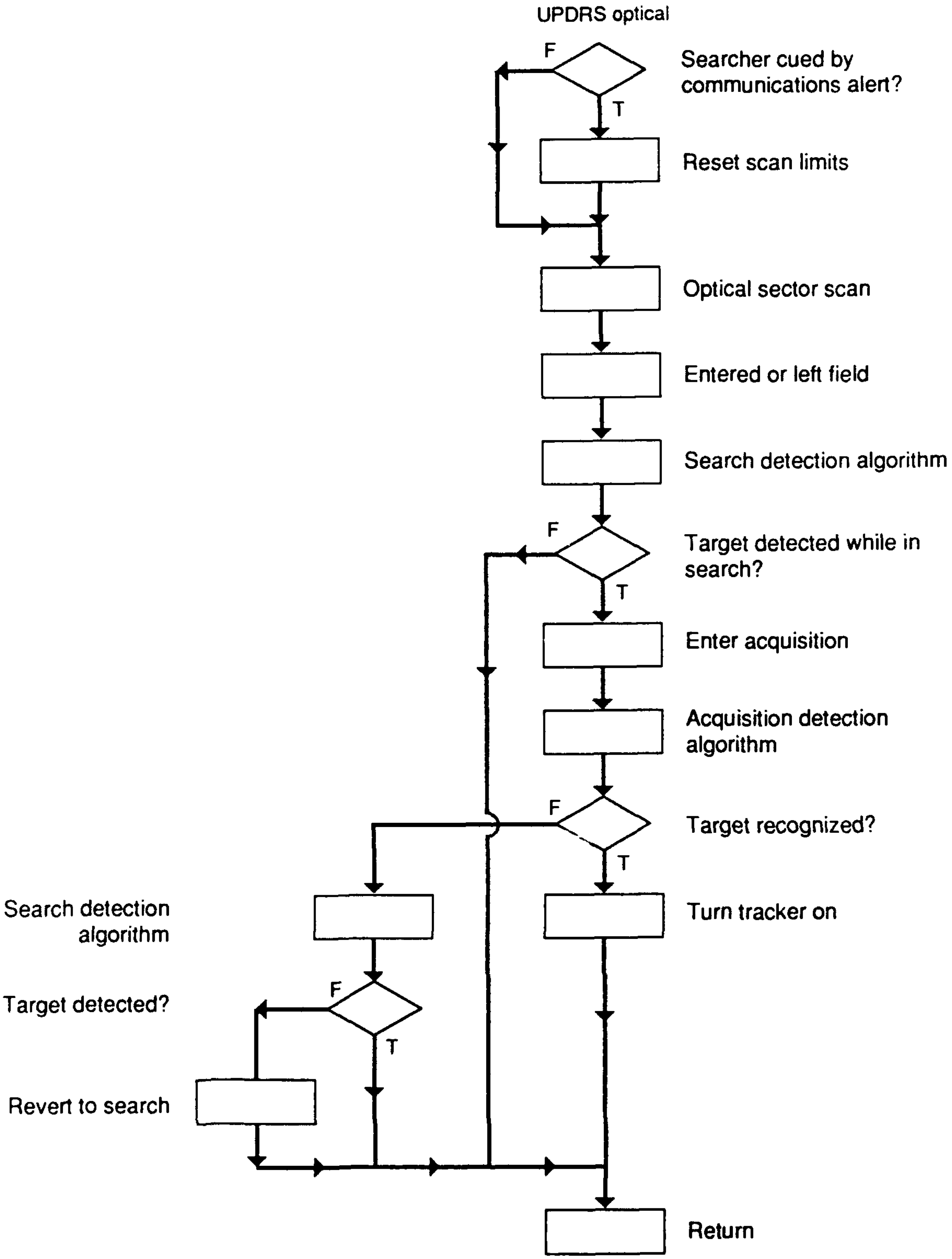


Fig. A.6—Update optical search and acquisition flow chart

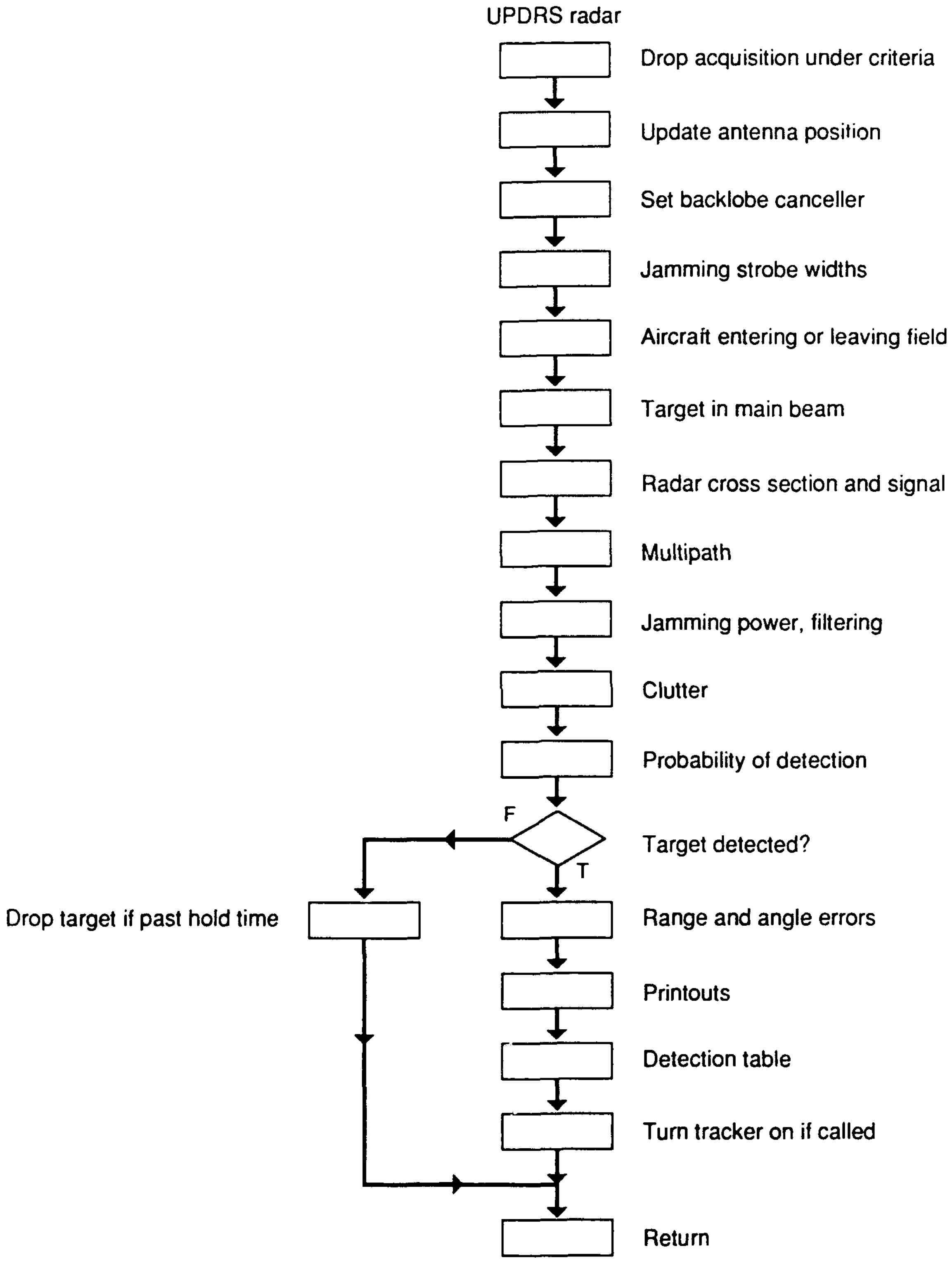


Fig. A.7—Update radar search and acquisition flow chart

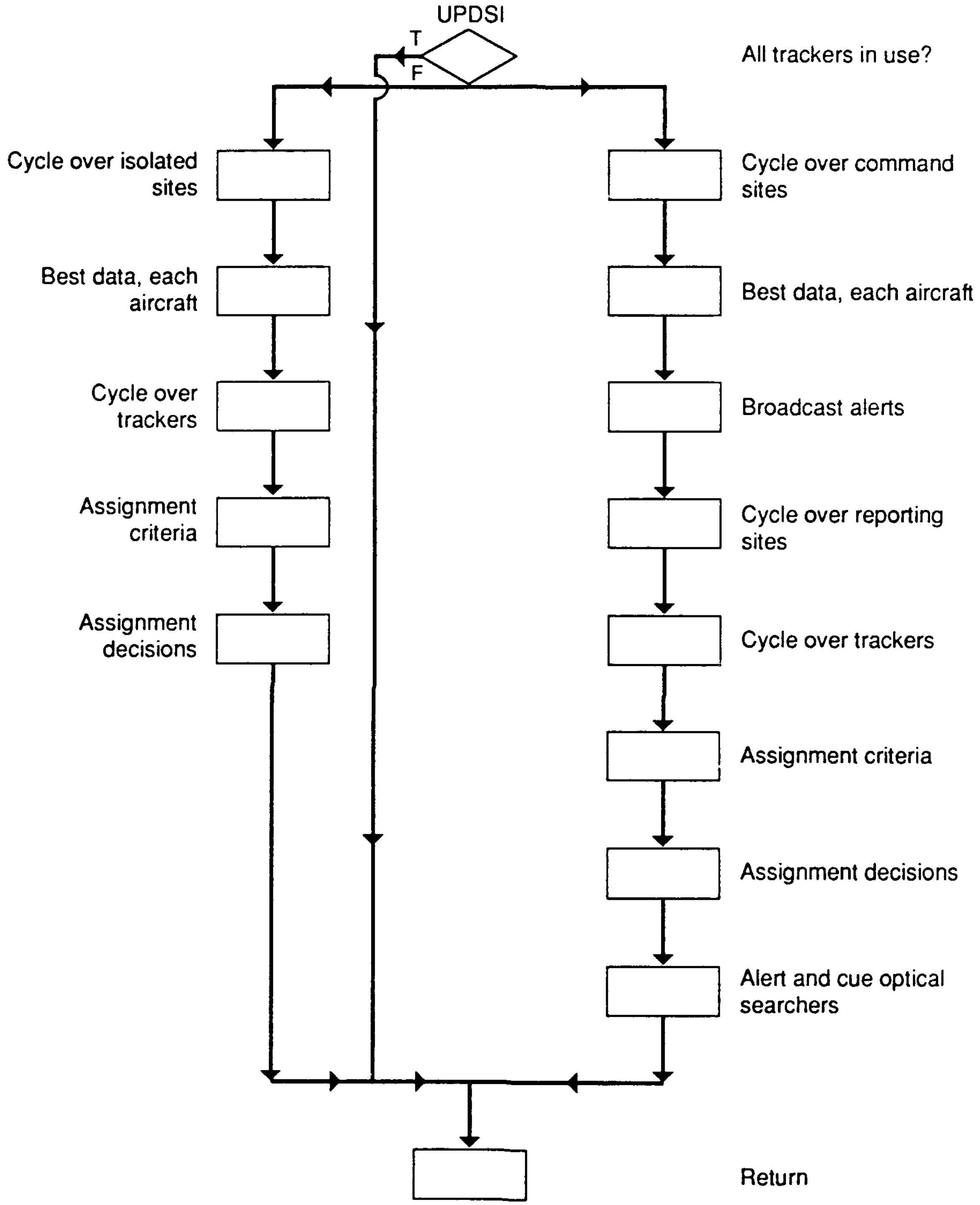


Fig. A.8—Update sites (UPDSI) flow chart

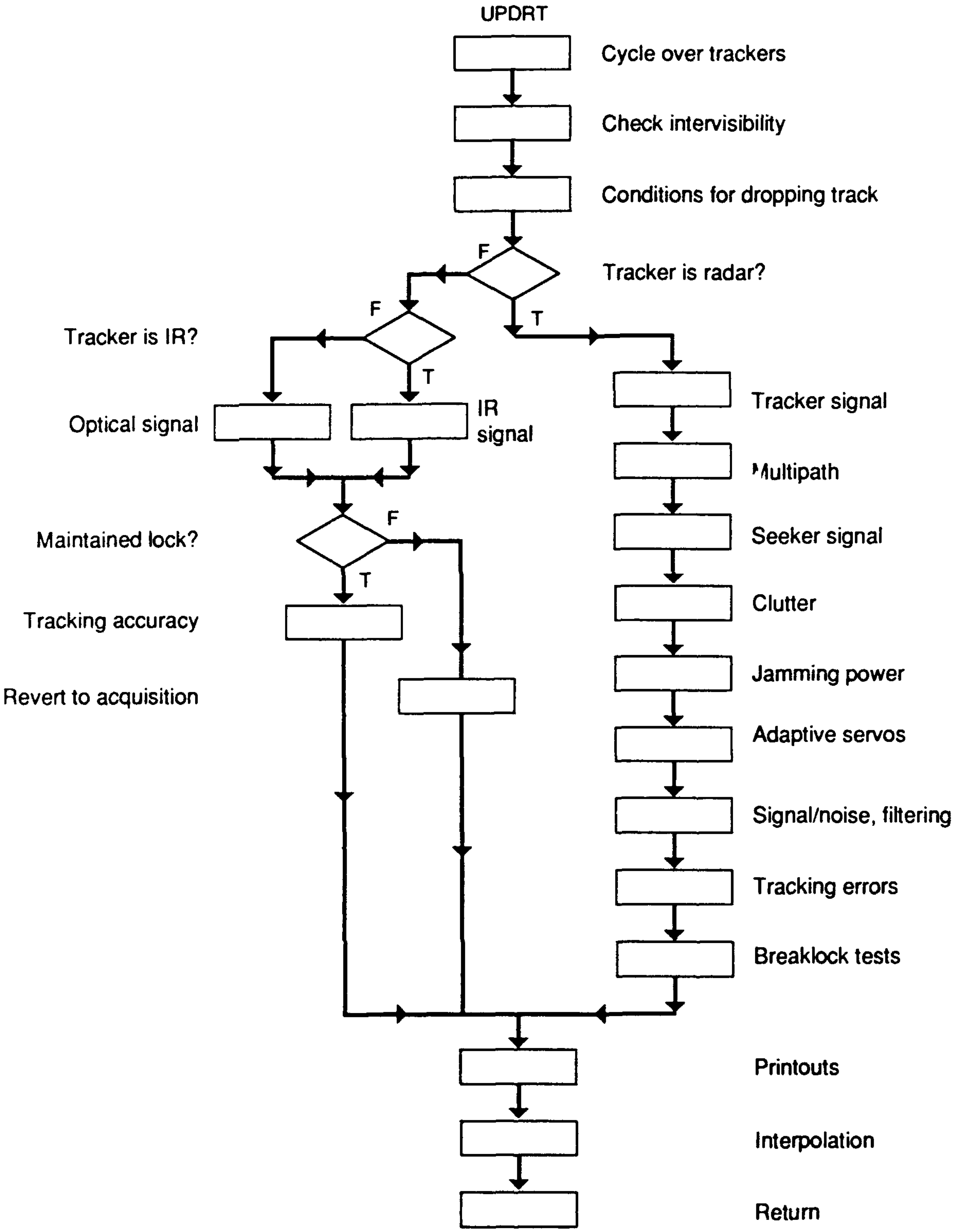


Fig. A.9—Update trackers (UPDRT) flow chart

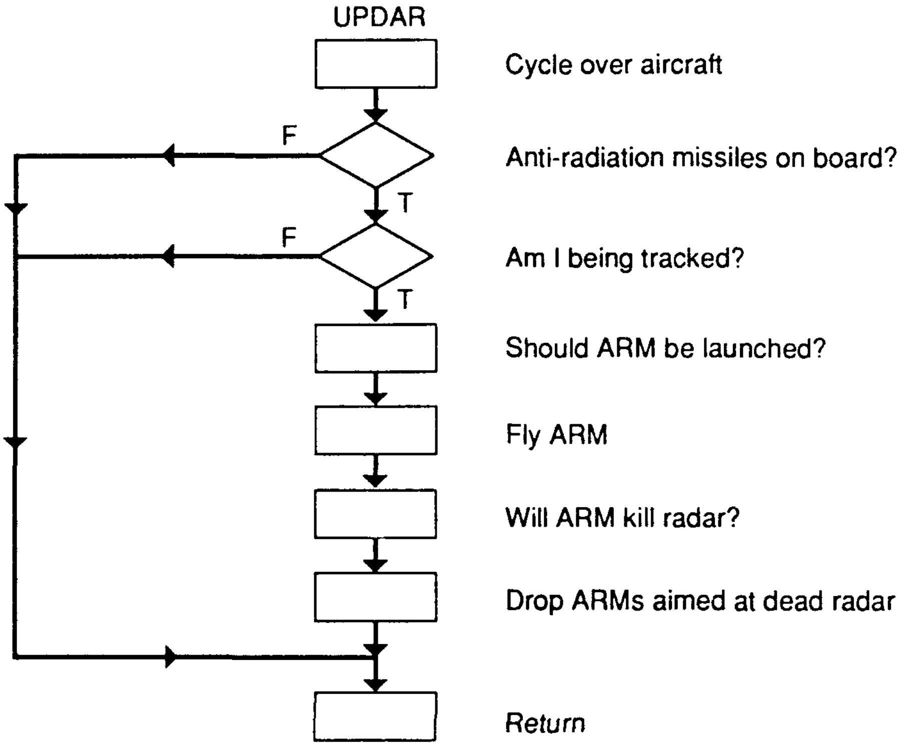


Fig. A.10—Update anti-radiation missiles (UPDAR) flow chart

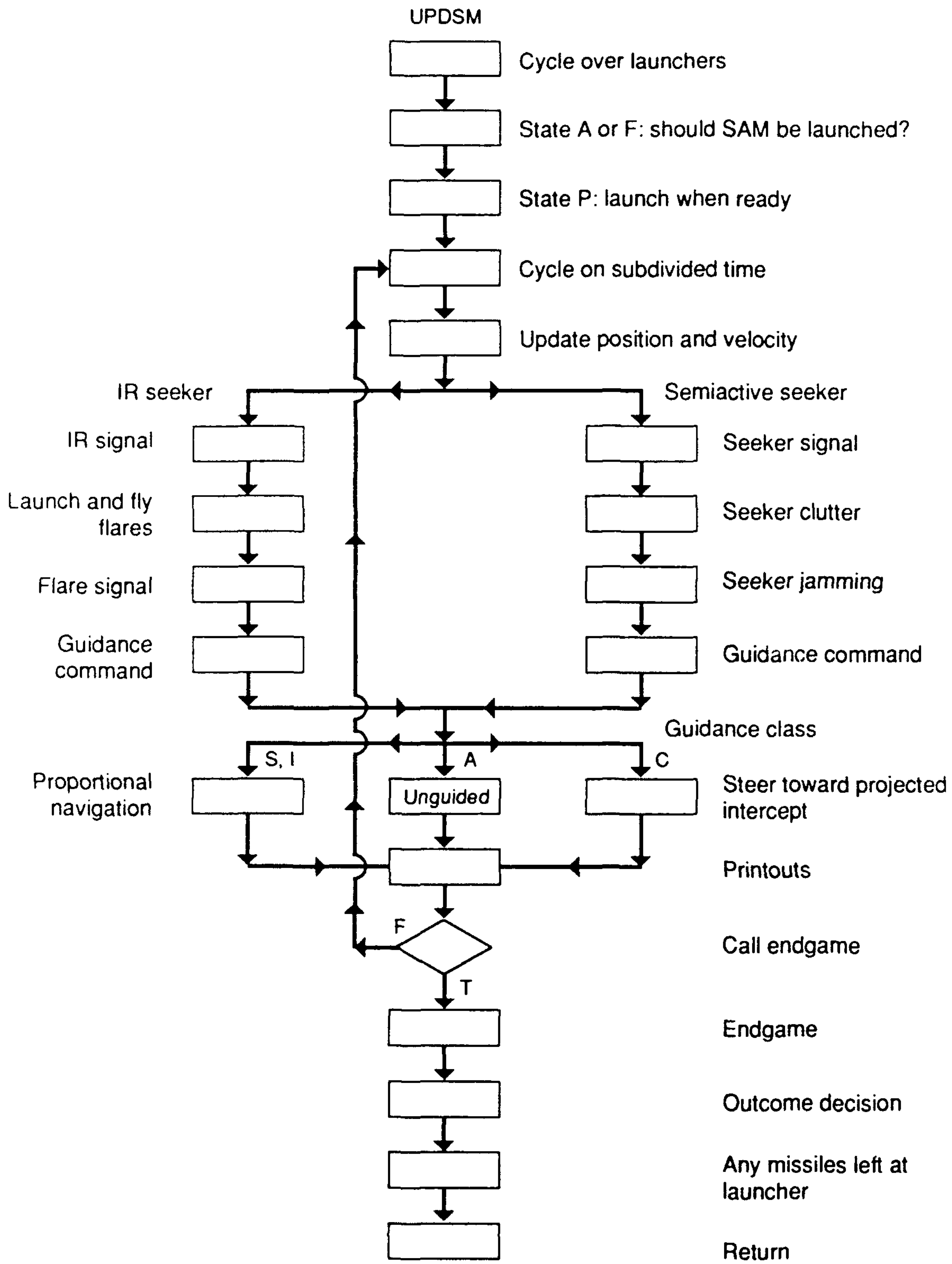


Fig. A.11—Update surface-to-air missiles (UPDSM) flow chart

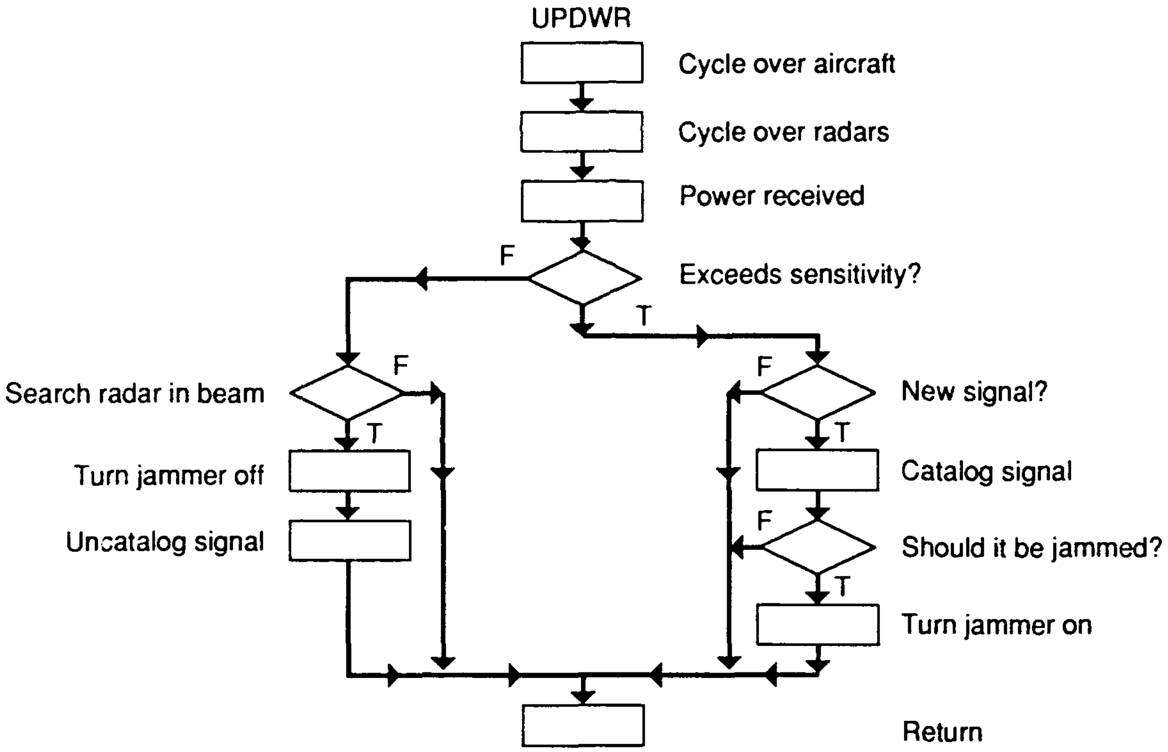
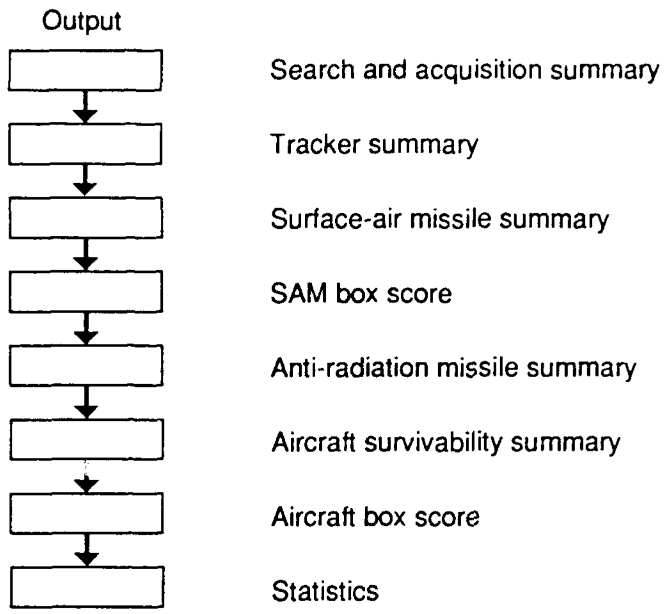


Fig. A.12—Update warning receivers (UPDWR) flow chart



UPDMC

Reinitialize all variables except random number generator
Recalculate availabilities if required

Fig. A.13—Output and Monte Carlo (UPDMC) flow chart

REFERENCES

1. Hubbs, J. E., *The Jamming Aircraft and Radar Simulation (JARS) Program*, Naval Warfare Analysis Division, Johns Hopkins University Applied Physics Laboratory, Report NWA-83-017, Laurel, MD, May 1983.

2. Hubbs, J. E. and G. W. Loughran, *The Modified Jamming Aircraft and Radar Simulation (JARSM) Program*, Naval Warfare Analysis Division, Johns Hopkins University Applied Physics Laboratory, Report NWA-84-027, Laurel, MD, May 1984.

3. Sollfrey, W., *RJARS: RAND's Version of the Jamming Aircraft and Radar Simulation*, The RAND Corporation, N-2727-AF, Santa Monica, CA, December 1988.

4. Skolnik, M. I., *Radar Handbook*, McGraw-Hill Book Co., New York, NY, 1970.

5. Barton, D. K., *Radar Systems Analysis*, Artech House, Inc., Dedham, MA, 1976.

6. Baty, R. S. et al., *Enhanced Surface-to-Air Missile Simulation (ESAMS) Computer Program - Analyst Manual, Basic Methodology*, Flight Dynamics Laboratory, Air Force Wright Aeronautical Laboratories, Air Force Systems Command, Wright-Patterson Air Force Base, OH 45433, Auigust 1988. Report prepared by Booz-Allen & Hamilton, Inc., Report AFWAL-TR-88-3051, Beavercreek, OH.