COMPUTER AND ANALYTIC MODELS
OF
FIGHTER INTERCEPT CAPABILITY

A Thesis submitted for the degree of

Doctor of Philosophy

in

the University of London

by

Geoffrey Stuart Donovan

(Royal Holloway College)

ACKNOWLEDGEMENTS

This research was supported by the Ministry of Defence and Scicon Consultancy International Ltd., and most of the work was carried out while the author was a staff member at the Defence Operational Analysis Establishment and a consultant with Scicon.

It is essential that I thank my wife, Barbara, for her perseverence throughout the development of the thesis, and express my gratitude to my supervisor, Andrew Sheer, for his continuing guidance. Special thanks are due to Trevor Lord and Gerry Lorimer of DOAE, for their advice on both methodology and fighter operations, and to Peter Grainger for his substantial contribution to the development of the radar and clutter routines. Finally, I must record my indebtedness to Professor Coulter McDowell, Ken Bowen, Professor Martin Beale and Tom Pratt for their considerable encouragement and support.

COMPUTER AND ANALYTIC MODELS

OF

FIGHTER INTERCEPT CAPABILITY

by

G S Donovan

This thesis describes a simulation model of fighter operations, and sets it in the context of a hierarchy of defence models. The fighter model is then applied to two problems which developed from an assessment of the contribution of sensors to the air defence of the UK. These concern the influence of raid indirect routing and sensor information accuracy on the intercept capability of fighters scrambled from ground alert. Mathematical models of these two aspects of fighter operations are also developed; this dual approach ensures that the studies are both analytically understandable and operationally acceptable.

Examples are given of the number of fighters from a single base which could intercept a concentrated (point) raid against a single offset target. In the case of raid indirect routing the main variables are the angle of the incoming feint track and the warning distance. In the analysis of the effect of sensor information errors the main variables are the actual position of the raid when warning is given and the errors in raid coordinates, heading and speed due to degradation of the warning system. In both cases of indirect routing and information errors alternative scramble and control procedures are considered.

The work described in this thesis is a step towards an attempt to determine the more favourable procedures which a fighter force might adopt in the face of subtle raid tactics and degraded sensor performance. It also provides an illustration of the interplay between mathematical methods and computer models in the analysis of fighter operations.

CONTENTS

(Detailed contents of Parts 1 - 4 overleaf)

CONTENTS OF PART 1 - OVERVIEW

CONTENTS OF PART 2 - THE FIGHTER MODEL

CONTENTS OF PART 3 - A MATHEMATICAL ANALYSIS OF RAID INDIRECT ROUTING

CONTENTS OF PART 4 - THE EFFECTS OF SENSOR INFORMATION ERRORS ON FIGHTER INTERCEPT CAPABILITY

APPENDIX A OF PART 4 - MATHEMATICAL ANALYSIS OF SENSOR INFORMATION ERRORS

PART 1

OVERVIEW

CONTENTS OF PART 1 - OVERVIEW

1. INTRODUCTION

This thesis was initiated when the author was a staff member of the Defence Operational Analysis Establishment (DOAE). It describes the development of a simulation model of fighter operations in defence of the United Kingdom - the DOAE fighter model. The model is then applied to two problems in support of an assessment of the contribution of sensors to air defence. These problems concern the effects of raid indirect routing and sensor information accuracy on fighter intercept capability. Mathematical models of these problems are also developed, in order to provide a broad grasp of the interactions between the important variables. The thesis is divided into four parts as follows:

Part I provides an overview of the whole thesis. This includes a summary of the context in which the fighter model is set, the aim of the model, the level of detail within it and its structure. The operational problems to which the model is applied are then described, and the mathematical models of raid indirect routing and sensor information errors are outlined. Finally the conclusions of the thesis are presented, both on the interplay between computer and mathematical models and on the influence of deceptive raid tactics and sensor performance degradation on fighter intercept capability.

In Part 2 of the thesis a detailed description of the fighter model is given. An introductory section provides a non-technical survey of the structure and assumptions of the model. This is followed by a detailed description of each of the model subroutines, in alphabetical order. An illustrative model input data stream is also included.

Finally, Parts 3 and 4 of the thesis comprise the detailed mathematical treatment of the effects of indirect routing and Ground Control sensor information errors on the intercept capability of fighters scrambled from ground alert.

2. THE DEVELOPMENT OF A FIGHTER MODEL SIMULATION

The fighter model was developed as part of a suite of air defence models at DOAE. The three principal computer simulation models for analysis of the air defence of the U.K. landmass are the Air Defence Ground Environment (ADGE) model, which represents the processes of gathering information and controlling the air defence system; EVA, a model of surface-to-air missile (SAM) air defence systems, and the fighter model. The essential components of the three models are defined by Jordan (1974); a description of the ADGE model is given by Donovan and Sands (1977), while the SAM air defence model is described by Thornton (1977, 1978).

Jordan (1974) also outlines the links between the ADGE, SAM and fighter operations models. Thornton (1977) puts the SAM model into an operational context. The operational setting of the ADGE model is given by Coucill (1975a, 1975b) and Coucill and Grainger (1976); ADGE operational techniques are summarised in the Central Tactics and Trials Organisation (CTTO) Tactics Manual (1978).

3. AIM OF THE MODEL

The fighter model has been developed in order to contribute to DOAE studies. These vary widely in their nature and scope, but may be summarised as concerned with defence strategy, UK and NATO defence policy decisions, and with resource allocation, especially of equipment, within the Defence Budget. Studies can range from examination of overall force structure to precise questions about the specification of particular pieces of equipment. As outlined by Lord (1978), examples of such studies are those which seek to determine:-

a) quantitative comparisons of competing procurement options;

b) tactical deployments to make the best use of specified equipment;

c) significant gaps in available or proposed equipment.

The prime difficulty in providing techniques with which to address these problems lies in matching the amount of detail required to achieve operational acceptability on the one hand and to preserve analytical understanding on the other. While the fighter model is essentially a 'general purpose' model, it was constructed with the investigation of a number of well-defined problems in mind. As described by Grainger (1975a) the most pressing category of problems is concerned with the effects of electronic counter-measures (ECM) when used in a large-scale conflict between technically sophisticated forces. In the absence of expensive large-scale trials, a computer model offers the best opportunity for deciding whether ECM provides significant protection for offensive forces, and whether the ECM should be directed primarily against ground-based or airborne surveillance radars, or against the fighters' airborne intercept (AI) radars (Hawes, 1975).

Grainger mentions some other problems for which the fighter model would be a useful tool. These include the assessment of the value to the UK and NATO of early warning means such as NIMROD Airborne Early Warning (AEW) and AWACS (Klass, 1974; Boyle and Furlong, 1977; Boyle, 1979) and the determination of suitable patrol positions for these aircraft (Watts, 1976). A further class of problems is concerned with operations or tactical aspects, such as examining the merits of maintaining continuous airborne fighter patrols (Combat Air Patrols, or CAPs) as opposed to relying on fighters scrambled from ground alert (DOAE, 1974; Bourne, 1975; Easams, 1976a); comparing fighter reattack policies (Grenader, 1968; Goda and Hanks, 1971; Burdon et al, 1977); and examining the value of supersonic interceptions (British Aerospace, 1980; Davies and Van Dijk, 1975).

Further areas of potential application of the fighter model are in the fields of weapon and equipment procurement, and in determining the most appropriate aircraft, or aircraft configuration, for the air defence role under various scenarios (Rose and Hogsflesh, 1974; Chief Scientist (RAF), 1974; Midgley, 1975; Chief Scientist (RAF), 1975). The model may also be a useful tool in naval air defence analysis. With suitable input data, the model could be used to investigate the effectiveness of land-based maritime patrol aircraft and fighters, and also carrier-based fighters. The contribution of tanker aircraft to fighter intercept capability can be assessed, as can the comparative value of the two types of CAP commonly associated with air defence of naval forces - barrier CAPs (e.g. across the Greenland-Iceland-UK Gap) and force CAPs, in which fighters patrol very much closer to the naval forces. Previous studies in these areas include those by Sutcliffe (1974) and Bourne (1976), while Forder (1979) carried out an integrated maritime air defence analysis, representing all the UK naval air defence systems planned for the 1990's. Rasmussen (1978) surveys the threat from the Soviet naval air force.

The fighter model may be used in analyses of the air defence of Western Europe. Numerous studies of aspects of NATO air defence have been carried out - see, for example, Boulton (1972), Facey (1975) and Andrews (1979). Meller (1975) and Gullick (1979) provide some detail on the Soviet air threat to Western Europe. A more specialised area of application is in the assessment of battlefield fighter effectiveness (Donovan, 1976a). This is concerned with the air defence, by fighters, of ground forces at or near the forward edge of the battle area (FEBA), against enemy ground attack aircraft. Both visual and radar detections are important in battlefield scenarios, where the duration of enemy aircraft penetration behind the FEBA may be very short (Whitaker, 1970; Cochram, 1971a, 1971b). Particular parameters of interest in such operations are the proportions of ground attack aircraft intercepted before and after weapon release respectively (Cocks, 1976).

The model can represent alternative tactics on both the attacking and defending sides, and insofar as tactics are explicitly represented, details are given in Part 2. It is not the intention of the model, however, to define the tactical behaviour of a force, but rather to leave tactics as much as possible a function of model input data. A long term air defence strategy is not represented; in particular, the recovery, turn-round and serviceability of fighters is not simulated. These are 'campaign' level factors (see Chapter 5). As such, the fighter model output can be used in an analysis of the interactions between fighter intercept capability, raid frequency and attack effectiveness, and fighter availability. The factors which contribute to fighter availability have been studied extensively - see, for example, Collins and Guthrie (1963), Guthrie and Means (1968), Burke (1974), Midgley (1974) and Hall (1974).

The fighter model is applied in this thesis to two specific studies of fighter intercept capability, concerned with raid indirect routing and sensor information errors. The interpretation of the model output for these two studies led to the development of the mathematical models presented in Parts 3 and 4 of the thesis, and is discussed later.

4. MODEL SCENARIO

Many of the characteristics of fighter model scenarios are defined in the input data, so that the model may be used to study fighter effectiveness in different spheres of combat. We shall concentrate here on the primary scenario - air defence of the UK mainland. This scenario envisages the incursion into friendly airspace - namely the UK Air Defence Region (UKADR) - of a sequence of enemy aircraft raids, of variable strength, formation, composition, speed, altitude, timing and direction of attack. Details of the likely threat to the UKADR in the 80's and 90's are given by Bloomfield, Constable and Spencer (1977) and by DSTI (1975a).

Both raid tracks and raid targets are represented in the fighter model, although the effects of the attacks by the raids on their targets are not - this being a 'campaign' level factor. Raids follow pre-planned routes to their various targets, or act as 'decoys', forcing premature and wasteful fighter scrambles. The specification of targets on the UK mainland for attack by Warsaw Pact aircraft is always a thorny issue. Donovan and Stead (1978) list some of the most likely UK military targets, under a number of headings - UK strike bases, UK maritime targets, USAF bases (permanent and reinforcement), RAF dispersal or forward operating bases, fighter bases, command and control centres and radar installations. (It becomes clear from the geographical distribution of these targets - East Anglia, S. England - that the Soviet Tactical Air Force is a threat to many of the UK military bases, in addition to their Long Range Air Force. Panyalev (1976) shows typical combat radii for the SU-19 'Fencer', based on forward deployment to East Germany).

Having defined the raid tracks and their targets we next consider the early warning and reaction capability of the air defence system. This is determined by a number of factors, the first of which is the ability of the surveillance system (ground and airborne) to detect raids at long ranges (see Borgart, 1977). The ADGE model can be used to simulate in detail the initial detection of a raid by ground or airborne radars. This, in turn, provides input to the fighter model in the form of contours, specified relative to a common coordinate system, whereby the times at which raids are first deemed to be detected within the fighter model can be determined. The initial detection position can also be parameterised in the model, and sensitivity analysis carried out to determine its significance in any given scenario.

The effectiveness of ground-based surveillence radars is severely limited against low-flying raids. In order to compensate for this gap in cover, airborne early warning aircraft are being introduced - NIMROD in the UK, AWACS in Western Europe (Ayker, 1975; Driessen and Taal, 1975). Numerous studies have been carried out into the detection and tracking of low altitude targets - for example, Gullick (1974), Macdonald (1974) and Litfin (1974). The model input data on estimated target position, track and raid strength, together with distributions for the errors on these estimates, may reflect information derived either via ground-based or airborne surveillance radars. Hence the fighter model can be used to investigate, at the sector level of operations, the contribution of AEW or AWACS to the command and control system.

In addition to the initial detection of a raid, its identification as hostile also takes place outside the model. Vaughan and Virnelson (1963) and Boyle (1977) have examined some of the problems concerned with the identification of hostile aircraft. Similarly, the control and coordination of air defence resources is an activity best simulated within an ADGE model. In particular, the allocation policy within the fighter model is concerned only with matching the immediate threats, rather than directly representing some overall strategy. Nevertheless, many of the interactions between the command and control system and enemy raids can be represented indirectly in the fighter model, without having to simulate the Air Defence Ground Environment, by appropriate choice of the relevant input parameters. For example, the vulnerability of communications links to jamming (Brown and Schemel, 1975) can be modelled in terms of degraded timeliness and quality of raid track and positional information passed to the fighter controllers. The data rate (i.e. frequency of transmission of information) may also be reduced by communications jamming.

Having detected and classified an enemy raid as a potential threat to the UK, scramble orders are issued to the appropriate fighter bases. The model simulates the delays in reporting target information, and the various readiness levels at which fighters are held. Fighters may be scrambled before detailed raid track information is available, in which case they fly to pre-planned patrol positions. When raid track information does become available, via ground or AEW radars, fighter AI radars or visual identification, fighter interception courses may be set up. The model represents delays in reporting target information and its accuracy (position, speed, heading), for the different sources of information. Fighters may refine their interception courses after they achieve target detection.

Enemy raids may also be supported by ECM - specialist escort jammers, self-screening jamming pods and stand-off jammers (Reed, 1974; Sundaram, Loomis and Eustace, 1976). Jamming may be directed either against the fighters' radars or against ground and airborne surveillance radars (or indeed against radar-controlled missiles launched by the fighters). The model represents both barrage (continuous) jamming and responsive jamming of fighter radars. In the latter case intermittent jamming signals are transmitted in response to incident 'threat' radar signals, dependent on the frequency and strength of these signals. DSTI (1975b) provides considerable detail on Soviet electronic warfare capability. For each airborne fighter and raid, at each radar scan, the model calculates a number of signal strengths - target radar return, perceived jamming strength, noise and clutter intensities - from which are derived a number of signal-to-noise ratios. These may then be compared with appropriate threshold values to determine whether a fighter has detected a target, and if it currently possesses range information on that target. The outcome of these comparisons then determines the subsequent behaviour of the fighter on its attempted interception. (The model may run in two modes, deterministic and stochastic. In stochastic mode a probability of detection is also calculated and compared with a uniformly distributed random variable).

In the timescales envisaged for fighter model scenarios, the Tornado or MRCA ADV (Multi Role Combat Aircraft, Air Defence Version - Gilson, 1978) and Phantom aircraft provide the primary RAF area defence capability. This influences the choice of signal-to-noise ratios used to simulate the fighters' avionics systems. Royal Aircraft Establishment reports (1974a, 1974b) provide data on Tornado and Phantom avionics system performance, in both clear and ECM environments. In particular, they provide numerical estimates of the various threshold values with which the fighter-target signal-to-noise ratios should be compared.

5. LEVEL OF DETAIL IN THE MODEL

The air defence suite of models must be capable of demonstrating the contribution of the air battle to the total land war in Continental Europe. Studies of total war can be handled by aggregated models of naval, air and land battles, which draw their inputs from more refined models and models of the separate campaigns which contribute to the outcome of total war. A useful categorisation of model hierarchies is given by Nowottny (1980). This was created in order to establish a methodology for derivation of naval vessel fighting characteristics, and is extended here in order to illustrate the role of the fighter model within a total hierarchy of models. Seven levels are shown, together with examples of typical model components at each level.

WAR	- Land battle in Europe; contribution of air operations to the land battle.
CAMPAIGN	- Contribution of fighters to the air battle.
FORCE	- Total fighter force within an air defence sector, one cycle of operations.
PLATFORM	- Individual fighter; includes total fighter characteristics (endurance, speed, acceleration, navigation, survivability, etc.).
TOTAL WEAPON	- Radar-controlled missiles; infra-red missiles; guns.
INDIVIDUAL WEAPON/SENSORS	- AI radar; ESM; missiles; fire control computer, etc.
COMPONENT	- Radar detection threshold; specific tracking algorithm etc.

Hierarchy for Military Modelling

It is essential that, at the start of any study or modelling exercise, the principal level of operations and principal elements in the analysis (as between, for example, the levels listed above) should be identified. Higher levels (of wider implication) than the one of particular interest impose criteria or constraints on the analysis, while lower levels (of greater detail) contribute input data. The process of identifying the main level of operations should include the determination of the principal measures of effectiveness. The fighter model is essentially a force model. Higher levels, which impose criteria on the model scenarios, include air defence battles over several days, an entire air campaign (including defence suppression, counter- air operations and offensive support in the Central Region as well as air defence of the UK) and, ultimately, the whole sea/land/air war. Lower levels, which provide inputs to the model, include the details of fighter/bomber engagements, the kinematics of fighter flight and the process of providing early warning (Lord, 1978).

To answer the sort of questions posed in the previous chapters, a model is required in which all fighters controlled from one point are individually represented. The air defence sector is thus the lowest level at which the model may realistically be set. (The U.K. Air Defence Region is composed of three such sectors.) At this level substantial simplification is necessary, and the model does not attempt a detailed representation of, for example, air-to-air combat (dogfights) or exact missile flight paths. Similarly, the representation of fighter aerodynamics is highly simplified.

A number of aims implicit in the development of a simulation such as the fighter model influence the level of detail within it. These include the following:

represent entities, events and their interactions as realistically as possible;

minimise the computer running time and program storage space;

make the preparation of model input simple;

make the interpretation of model output simple;

Allowing for an emphasis on areas of particular relevance, the degree of sophistication of the model is also designed to be consistent over the entitites and interactions represented within the model, and consistent with a model at the 'force' level of fighter operations.

The input data required to run the fighter model is explained in full in Part 2. In any particular application of the model many of the data items are not relevant. Many of the input variables may also be parameterised, in order to conduct sensitivity analyses. Nevertheless, in the assessment of the effectiveness of, for example, a specific aircraft or aircraft component, a particular threat or a new enemy ECM technique, more detailed weapon system models may be necessary, or more detailed analytical studies undertaken, in order to derive the relevant input parameters for the fighter model. With such a hierarchical approach to computer and analytical modelling, the influence of important parameters - fighter response time, fighter radar capability, enemy strength, threat patterns, raid timing, enemy jamming capability, etc. - can be measured in the fighter model, at the 'force' level of operations. Similarly, the fighter model itself can provide more highly aggregated information suitable for input to models at 'campaign' level of operation. At DOAE these include the Air Campaign Model (Douch, 1975) and a similar model in support of an air campaign game (George, 1979). Both Douch and George describe the way in which various tasks such as air defence, counter air operations, interdiction and close air support, and the allocation of aircraft to these tasks, are represented in campaign models.

Campaign models, in their turn, may be used to provide input data for the highest level in the military model hierarchy - war games or computerised war games (Shephard, 1963). The methodology by which the effects of air campaigns on the land battle are assessed is described by Lord, Donovan and Lee (1976), Donovan (1976b) and Dyer (1976). A major element in the calculation of the effectiveness of air campaigns is the role of aircraft attack effectiveness - see, for example, King (1975a, 1975b). The NATO Deployment Model (Dare, 1972) is a computerised war game, requiring detailed assessment by models such as the fighter model for its calibration. A number of campaign or theatre-level air battle models have been developed in the US; these include AIR-2 (Honningstad and Kerr, 1973), NEWAIR (Harreschou and Kerr, 1975; Roros, 1976) and IDA Tactical Air Model (Lowell et al, 1979). The bibliography by Berg (1980) lists a number of air battle and air defence simulation models developed on behalf of US defence agencies.

The process of aggregating results from detailed models to provide calibration and input data for higher level models is well developed in the field of land battle modelling. The two primary land battle models used at DOAE are the Battle Group Model and the Corps Model (Beare et al, 1974; Witts, 1974). The models interface by condensing results from the Battle Group model of combat, involving several weapon types on either side, into a form in which the performance of each force as a whole can be represented. In particular, two related problems are solved within this interface:-

a) a technique is developed by which the strength of a battle group can be calculated from knowledge of its component weapons;

b) a measure of the ability of one battle group to cause attrition to another is derived.

These two measures are fundamental to the Corps model assessment of battle. The Battle Group and Corps models, and the progressive refinement of the interface between them, have been well documented (e.g. Dare and James, 1971; Richardson, 1976; James, 1976; James, 1977; Richardson, 1977; Richardson and Dunkerley, 1977). Howes and Thrall (1973) provide a similar methodology for constructing a system of weapon weights from weapon effectiveness tables, obtained from detailed simulation.

As a final example of a hierarchical approach to model-building, Aris (1977) provides a hierarchy of techniques in a field of mathematical physics. This includes an analysis of the level of detail and domain of applicability for six models of a diffusion process.

6. OTHER MODELS OF FIGHTER OPERATIONS

A number of quite substantial simulation models of fighter operations have been developed over the last few years. No existing model appropriate to U.K. air defence analysis represents ECM with fighter operations at the level of detail required by DOAE. Two of the models closest to that required for DOAE studies were developed by EASAMS Ltd., and examined in detail by Grainger (1975b). These were the Multiple Engagement Simulation Model and the Fighter Deployment Battle Model (FDBM) (EASAMS, 1973). Shipman (1974) used the FDBM for a study of threat evaluation and fighter allocation, examining simple tactical and operating procedures. EASAMS has also written a number of models for more specific applications. EASAMS (1977) is a study of the Phantom interception system in an ECM environment, while EASAMS (1976b) is an interception model created in order to assess the fire control and radar systems of an interceptor up to the point of weapon release. This represents in detail the Phantom F4-M with AWG-12 radar. An interesting feature is a simple pilot model, modifying steering errors produced by the fire control computer.

Another model which simulates both fighters and enemy aircraft in too much detail for our requirements is given by Welp, Brown and Rea (1975). This is a Monte-Carlo simulation of opposing aircraft flights during an entire mission, including communications, navigation, refueling, target detection, target identification, attack and weapon detection. It is also worth mentioning the powerful modelling tool COMO-III, developed at Shape Technical Centre (STC) for NATO defence analysis (Mann, 1967; Happel and Mulders, 1970; Dockery, Leiser and Aitken, 1976). Rather than a model, COMO is a simulation modelling system, permitting direct interaction between the decision-maker and the (critical-event) simulation.

It is useful to note some of the more detailed models and analyses which could provide data input to the fighter model in particular studies. For example, missile flight time and lethality, as a function of range, missile/target geometry and ECM conditions, are model input parameters. EASAMS (1969) describes a computer model of the Sidewinder infra-red missile, while EASAMS (1971) studies the homing ability of the Sparrow semi-active radar missile. Bok (1973) predicts Sparrow lethality against Soviet fighter aircraft, while Lawrence and Cairns (1976) have developed a hybrid simulation of the Sky Flash missile and its radar environment. Schenk (1976) analyses the evolution of the Sidewinder family of air-to-air missiles, culminating in the AIM-9L 'Super-Sidewinder' - an infra-red missile with a head-on attack capability.

For a force level model, the fighter model goes into considerable detail in its radar detection routines - i.e. in the calculation of target signal strengths, enemy aircraft jamming strategies and consequent jamming signal strengths, and pulse and pulse doppler clutter levels. The report by RAE (1974b) indicates additional aspects of the responsive interaction between modern fighter radars and Warsaw Pact airborne jammers, together with fighter radar reversionary modes and detection thresholds. Forsyth and Penwick (1976) describe the SPEARS air warfare model in which several alternative detection criteria are presented.

The fighter model does not represent directly the tracking process which follows target detection, nor the generation of radar false alarms and the consequences of such false alarms. While it is felt that such detailed representation is more appropriate in lower level models, the modular nature of the fighter model allows the inclusion of routines such as automatic multiple target tracking (British Aircraft Corporation, 1975) and generation of radar false alarms (Skolnik, 1962; Bishop and Murray, 1975). Wong (1974) presents an analysis of the problem of correlating radar returns with aircraft tracks, given limits on aircraft manoeuvrability and acceleration.

The representation of visual detection and identification in the fighter model is elementary, with accurate visual detection assumed to occur when an enemy aircraft comes within a fixed distance from a fighter. This distance is specified in the input data, as a function of the fighter type. Seyb (1967) and Sparre (1971-72) have analysed in detail the mechanics of visual detection. Their methodology includes physical laws relating the contrast between a target and its background environment (e.g. meteorological visibility) and the target's apparent contrast, to the position of the fighter aircraft relative to the target. It also includes empirical relations between the eye's capability to detect a target in a single glimpse, the target's apparent contrast, its angular distance off the fighter pilot's line of sight, and the target's apparent size. They also refer to such factors as cockpit design, pilot search policy and enemy raid formation structure.

The fighter model contains a set of subroutines which simulate fighter re-attack sequences. In these routines the raid structure is represented in more detail than when calculating initial interception courses, for we are searching for subsequent targets to attack within a raid. As mentioned earlier, the model does not simulate the details of air-to-air missile firings, gun attacks and the associated attack and evasive manoeuvres, nor counter-attacks by the enemy bombers or possible escort fighters. In particular, fighter casualties are not represented. Air-to-air combat is an area of air operations which has been studied extensively. Anderson (1972) reviews methods of assessing attrition in air-to-air engagements, while Greene and Huntzicker (1967) discuss the mechanics and geometry of air-to-air engagements. British Aerospace (1977) studies the air combat capability of a particular aircraft (AST403), while Beecham (1972) and Vincent et al (1976) analyse the kinematics of air combat and the missile avoidance problem respectively. Welp and Brown (1974) assess a number of models of air-to-air combat.

7. MODEL STRUCTURE

Event Processing

The model is an 'event-based' simulation, as opposed to a 'time-stepping' simulation; any significant occurrence to an entity in the model is called an event for that entity. For example, events for enemy raids include track-change points and weapon-release; events for fighters include take-off, change of acceleration or fuel-consumption rate, and detection of target. At an occurrence of any event all relevant consequences of that event are considered and, in particular, any new events which are generated must be set up. The heart of the model is simply a list of all events and the data associated with each event - the time at which it is to occur, the type of the event and the entity to which it occurs. As each event occurs and its consequences are examined this list is amended as necessary, with new events being inserted and cancelled events deleted.

A number of methods have been proposed for storing and updating lists of events - see, e.g. Lave (1967), Nance (1971) and Babich, Grason and Parnas (1975). The method of linked lists is adopted in the fighter model, based upon the algorithm of Lambeth (1978). This method carries out the three functions of event scheduling routines - scheduling future events, executing events in the proper simulated time sequence, and updating the clock variable which represents time - with economy of computer processing time and list storage space. A demonstration of the advantages of linked list event-scheduling routines, in terms of reduced computer processing time, is given by Engelbrecht-Wiggans (1978). In the Lambeth algorithm, each event is placed in any free cell of an array, and has stored with it a pointer to the next later event. Events generated during a simulation run can be stored in cells vacated by events which have already been processed. Pointers to the next event and to the end of a linked list of free cells are held externally; a special value of a pointer indicates the end of a list. Removal of the current event from the list of events and placing its cell on the free list does not require large amounts of data to be moved in store; only the values of pointers are changed. This applies also to the insertion of a new event, and the deletion of an unwanted event.

More advanced event-sequencing algorithms, based on the idea of linked lists, are given by Wyman (1975) and Vaucher and Duval (1975).

Mode of Operation

Two versions of the model have been written, a stochastic ('Monte Carlo') version and a deterministic ('expected value') version. In the Monte Carlo model, whenever the value of a random variable is required, it is derived from the appropriate distribution using a random number generator. The uniform random number generator used in the fighter model is that described by Hall (1979), which is particularly appropriate to a computer with a 24 bit word length. A survey of uniform random number generators is given by Jansson (1966). Random variables from a normal distribution are obtained by the simple method of adding a number of independent samples from a uniform distribution. More efficient techniques for generating normal random variables are given by Box and Muller (1958), Tocher (1963) and Marsaglia and Bray (1964).

The results of the simulation are also random variables, and to obtain reliable information concerning their average values the program must be run many times with independent sets of random numbers. Currently the model uses a single seed from which are generated the several random number streams used in the stochastic version of the model. It is a minor extension to allow each random number stream its own seed (Mihram, 1976). Each replication terminates either after a predetermined simulation time or when the fighter force is annihilated.

The fighter model has only elementary statistics collection, namely estimates of the mean and standard deviation of random variables. There is considerable scope for implementing more powerful statistical techniques. The model is a terminating simulation, so that more techniques are available for statistical analysis of the model output than for 'steady state' simulations. Law (1980) examines a number of techniques for deriving confidence intervals for the mean values of random variables in terminating simulations. He also derives stopping rules for such simulations, i.e. rules which determine the number of independant replications necessary to produce required confidence intervals, with a specified relative or absolute width. (Law and Kelton (1978, 1979) survey fixed sample size and sequential procedures for constructing confidence intervals of steady state mean values). It would be valuable to assess the effectiveness of such techniques if incorporated in the fighter model.

In the deterministic model, on the other hand, each time a random variable is required, its use is replaced by a process intended to represent the 'expected result' of using the range of values which it can take. For this reason deterministic models are sometimes referred to as 'expected value' models. The model in its deterministic form has the advantage of being comparatively fast, and of being able to display the effect of small input changes which may be lost in the distribution of results from the equivalent Monte Carlo model. In its stochastic form, the model can offer guidance on specifically probabilistic questions, and it provides a means of calibrating the deterministic model.

We can assess the differences between the two versions of the model in terms of the combat cycle described by Smith (1968) - sightings, decision-taking, physical movement of combat units and firing.

Sighting - The two versions of the model differ in the representation of the fighter radar detection process, and in the estimation of raid position, speed and heading.

Decision-Taking - There is no difference between the two versions in the representation of decision-taking. All such processes in both versions are deterministic. To a large extent they are conditional upon input data, so that alternative decision processes can be investigated parametrically. These processes are not so complex that it was felt necessary to utilise interactive simulation techniques - see, for example, Ulvila and Brown (1978) and Modelski (1978). In such simulations all decision-making processes need not be formalised within model algorithms - players have freedom to make decisions during the execution of the simulation.

Physical Movement - In stochastic mode errors in the fighter heading are sampled from normal distributions, while in deterministic mode the model assumes a systematic heading error has been specified.

Firing - In stochastic mode the missile kill probability (P_k) is compared with a uniform random variable to determine if an enemy aircraft has been killed. In deterministic mode fractional numbers of aircraft are allowed and exactly P_k aircraft are killed after a successful fighter attack.

Model Output

We list here typical elements which may be included in the output from most runs of the fighter model. Firstly, it is always useful to record at an appropriate point in the output the important input parameters. These could include the following factors.

a) Whether the run was stochastic or deterministic.

b) If stochastic, how many replications.

c) The time-span represented in the run.

d) The number of raids, and the proportion of each raid consisting of escort jammers.

e) The number of fighter bases, with their names.

f) The number of fighter types, with their names.

g) Details of the timeliness and accuracy of the raid track information provided to the fighters by Ground Control.

h) Details of the fighter readiness levels.

i) Number of fighters scrambled on first detection of a raid; number of fighters scrambled when raid track information is available; ratio fighters: enemy aircraft required when the strength of a raid is known.

j) Name of the radar type in each fighter type; whether each fighter type has a data-link, and/or a track-while-scan radar.

k) Some details on the performance of each missile type.

The primary measures of effectiveness in the fighter model are the two factors which would be of immediate application in a 'campaign' level model - number of targets attacked successfully by the raids, and number of enemy aircraft shot down by fighters. In any particular study, statistics could also be collected on the following factors, under the broad headings of raid performance and fighter performance. Data may be presented for each fighter type and then accumulated or aggregated over all fighter types, or fighter bases.

a) Time from fighter take-off to start of first interception.

b) Co-ordinates of points where enemy aircraft shot down.

c) Total missiles fired.

d) Total enemy aircraft killed or damaged.

e) Fighter fuel/missiles still on board when forced to return to base.

f) Sortie duration.

g) Times at which enemy aircraft shot down.

h) Comparative success of fighters from different bases.

i) Reasons for failure of fighters to complete interceptions (e.g. raid track-change; raid passes through missile engagement zone; target already destroyed by another missile).

j) Effect on fighter intercept capability of ground radar or fighter radar tracking errors.

k) Comparison of the number and effectiveness of attacks by the various fighter types.

Verification and Validation

Considerable effort has been spent in verifying the model, i.e. in ensuring that it behaves as it is planned to behave under well-defined circumstances. The verification procedure includes four of the five model acceptability criteria developed by Hermann (1967) and summarised by Mihram (1972).

1. Internal Validity - using model replication, and holding model inputs constant, to determine whether the variance of the responses is not too large;

2. Face Validity - using subjective opinions regarding the surface impressions of the model's realism. (So-called 'military judgement' - extremely important if the model, and study results produced using the model, are to have credibility amongst the military and civilian study sponsors).

3. Variable-Parameter Validity - 'Sensitivity testing' to ascertain whether the effects of changes in the model's variables are compatible with comparable alterations in the modelled system.

4. Hypothesis Validity - examination of the sub-system relationships assumed within the model.

The fifth criterion - event validity - is now more commonly called validation (Fishman and Kiviat, 1967) - i.e. testing the agreement between the behaviour of the simulation model and the real air battle. Fortunately there has been no opportunity to validate the model in its principal scenarios. It may be possible to validate sub-sections of the model against operational trials - see, for example, Central Tactics and Trials Organisation (1975b & 1976), Parkinson et al (1978) and Clark et al (1978). Such validation is inevitably of limited value in a force or campaign level model; it is far more practicable and useful lower down the model hierarchy, at the total weapon, individual weapon or sensor, and component levels. Care must also be taken when extrapolating documented results on combat to the scenarios envisaged for the fighter model, namely intense combat between NATO and Warsaw Pact forces. For example Herzog (1975) quotes an exchange ratio of 334:5 for air-to-air combat in the 1973 Middle East war.

8. APPLICATIONS OF THE MODEL - RAID INDIRECT ROUTING AND SENSOR INFORMATION ERRORS

The fighter model has been used in support of a major study at DOAE (Study 250), which investigates the influence of sensors on fighter intercept capability - see Bloomfield and Spencer (1977). In particular, the study considers the effects of the type and number of sensors on the proportion of enemy aircraft raids which are interceptable, and examines the sensitivity of fighter capability to changes in the warning distance (i.e. the distance at which the raid is detected, and identified as hostile and a threat). Before we consider the specific applications of the fighter model it is worthwhile to summarise the various components and interactions involved in an air defence battle. Of these components, the contributions of sensors, together with fighters, surface-to-air missiles and command and control systems, are assessed by Lord, Bloomfield and Spencer (1978). Lord and Spencer (1977) provide an elementary examination of an air defence battle, in which they address the problem of how the influence of sensors on fighter intercept capability fits into the overall analysis of air defence capability.

a) Defence Components

The effectiveness of the air defence system - and indeed of the attacking aircraft - is measured in terms of the damage caused to the primary target system. The specification of this system is a major element in air defence analysis, since the loss or otherwise of primary targets is the major influence of the air defence battle on both the overall air campaign and the land and sea battles. Primary ground targets could include, for example, U.K. strike bases, naval dockyards, maritime headquarters, nuclear submarine bases, USAF bases (permanent and reinforcement), U.K. dispersal bases (i.e. forward operating bases), fighter bases, command and control centres, Fylingdales Ballistic Missile Early Warning System, air defence radars and sector operations centres (see, for example, Clark (1975)).

The principal aim of the air defence system is to reduce damage to these primary targets. Note that a significant degree of protection against enemy aircraft attack may be provided by passive defences. These could include, for example, the siting of strike aircraft in bomb-proof shelters and revetments; the camouflage of aircraft and ships; the use of dummy aircraft as spurious targets for low-level air-to-ground attacks and the dispersal of strike aircraft to the perimeters of their airfields.

The great weakness of passive defences is that they do not inflict losses on the attacking aircraft. In the face of passive defences only, enemy raids can continue flying sorties against the primary targets indefinitely, and whilst on a sortie they are not restricted in the amount of time spent over the target area in detecting, identifying and attacking specific targets. The role of surface-to-air missiles (SAMs) and fighters is to inflict losses on enemy aircraft, to force them to abort their missions and to minimise their attack effectiveness over the target area. These weapon systems can be classified as point defences and area defences. Point defences are short-range SAM systems - e.g. Rapier - permanently and securely located around important targets. By definition they should provide considerable protection to these targets, but not necessarily to any other targets. Area defences, on the other hand, comprise medium and long-range SAMs (Thunderbird, Nike) and fighters; these are flexible weapon systems, designed to provide protection over a wide area. Area defences also generally pose a significantly greater threat than point defences to transitting enemy aircraft. The location of the fighter bases can be a significant factor in determining fighter effectiveness. Lord and Spencer (1978) provide some examples of the comparative cost-effectiveness of fighters and point defences for given passive defences of offensive military airfields, over a spectrum of possible threats. They also consider the conditions under which an appropriate mix of fighter and point defences might be considered. The choice of area defence system is strongly dependent on the magnitude of the threat, the targetting policy of the attacker, and the efficiency of the warning and control system.

There are other considerations in the comparison of fighter and missile systems which are not examined here. Chief among these is the fact that future war is likely to be graduated, and our response in times of tension has to be correspondingly graduated. Fighter aircraft are admirably suited to provide such a response, as opposed to SAM systems. This problem is summarised by

Lord (1978), who raises the question of the balance between a 'dynamic' defensive system (founded on fighters and a complex communications, command and control system and oriented towards operations in peace and tension as well as war), and a 'static' defensive system (based on medium and short range SAMs and passive defences and of great deterrent value, but only likely to be operational in time of war).

The final air defence component to be described is the control system. Composed of sensors and a C^3 (Command, Control and Communication) system, it controls and co-ordinates the defensive response. The sensors may be Ground Control radars, airborne radars (Nimrod Airborne Early Warning aircraft), Continental Early Warning, fighter aircraft and SAM systems, and of course information received from strategically situated observers. Borgart (1977) discusses the various means of detecting attacking aircraft. The control system comprises ground-based and airborne command centres, with advanced data handling technology (Sutherland, 1976) to process the information received from the various sensors; this information may be of varying accuracy, dependent upon its source, and may only be received intermittently. The control system must also identify the raid as hostile (Boyle, 1977) and co-ordinate the defensive response - alert fighter bases and scramble aircraft; alert SAM systems; activate passive defensive measures (e.g. deploy dummy targets and disperse strike aircraft); re-deploy sensors such as AEW aircraft or transportable ground-based radars; activate ECCM responses in the face of enemy ECM, for ground-based radars (Central Tactics and Trials Organisation, 1975*a*), or airborne radars (e.g. Ayker, 1975; Driessen and Taal, 1975).

Bloomfield, Littlejohn and Spencer (1977) summarise some plans and concepts of operation for all these defence system components - fighters, SAMs, tankers, AEW, ground radars, main and dispersal aircraft bases, communications and command and control. Central Tactics and Trials Organisation (1978) provide more detail on operational techniques for the Air Defence Ground Environment (i.e. C^3 + sensors) in the United Kingdom Air Defence Region (UKADR).

b) <u>Threat Components</u>

Again, we begin the summary of the threat components with the primary target system. The enemy will attempt to destroy or disable these targets at the risk of the loss of his own attacking aircraft. These factors - damage caused to primary targets, enemy bombers and fighter-bombers lost, and enemy offensive sorties committed to attack U.K. targets which could have been used in support of the land battle elsewhere - are the means by which the outcome of the defensive air battle influences the overall air campaign, and hence the land battle. It is worth noting that the primary target system as perceived by the enemy may not necessarily coincide with that as perceived by the defence.

The damage to the primary targets is inflicted by attack weapons, and we assume here that these are carried or launched by attack aircraft. The attack weapons may be free-fall bombs, guided bombs or missiles, stand-off missiles (i.e. self-powered missiles, with or without a homing capability, launched at some distance from the targets), and rockets or guns. The number of attack aircraft required to destroy or incapacitate chosen targets is a major factor in assessing the likelihood of attack. The attack effectiveness, and therefore the damage inflicted on the primary target system, is balanced against the defence effectiveness, i.e. the number of aircraft which must be committed to enemy raids in order to reach the targets in sufficient strength for an effective attack, and the total enemy losses suffered in transitting to and from the target area and in carrying out the attack.

Finally, major contributing factors to attack effectiveness are the enemy's support measures and tactics. Support measures include, for example:

escort jamming aircraft, or self-screening jamming pods attached to the attack aircraft;

stand-off jammers to degrade ground-based surveillance radars or fighter radars;

escort fighters to protect the attack aircraft while in transit and over the target area.

Tactics could include:

spacing and timing attacks in order to saturate ground defences;

co-ordination of multiple attacks from all points of the compass;

varying the altitude of transit in order to reduce the warning time given to the defence system;

use of dog-legs or indirect routing by the raids in order to confuse the fighter defences.

Bloomfield, Constable and Spencer (1977) summarise the (conventional) air threat to the U.K. in the late 1980's, while DSTI (1975a, 1975b) provide a more detailed analysis, particular of Soviet ECM capability.

In order to analyse the effect of the major sensor parameters certain simplifying assumptions were made in Study 250. In particular, it was assumed that:

(i) raids attack their targets along direct tracks;

(ii) the information provided to the fighters by Ground Control is accurate.

Assumption (i) relates to the tactics adopted by enemy raids while travelling towards their targets. Assumption (ii) should correspond to an enemy attack in 'clear' conditions, or at least with ineffective ECM support.

In this thesis we address the problems raised by relaxing these particular assumptions, and investigate the influence of bomber tactical routing and raid tracking accuracy on the intercept capability of fighters scrambled from ground alert. The two operational aspects of this study are analysed independently. In studying raid indirect routing it is assumed that the information provided to the fighters by the sensors through Ground Control (GC) is accurate. In the analysis of the effect of GC information errors, it is assumed that raids fly along direct tracks to their targets. (Here GC information could also refer to information received via AEW aircraft. Clark (1975) has investigated the probability of a fighter detecting a raid as a function of the accuracy of AEW track determination).

The fighter model was used extensively in the early phase of the study of the effects of raid indirect routing on fighter intercept capability. Having defined a reasonable measure of effectiveness (namely, the number of fighter interceptions achieved), several graphs were produced which summarised the study results. These are essentially the graphs presented as Figures 2.13 - 2.16 in Part 3 of the thesis. It was evident from these results that, even with highly detailed simulation output, no clear explanation of the convoluted shape of some of the graphs was possible. (The explanation of the derivation of the fighter model results was complicated by the inadequacy of a numerical technique for calculating possible interception courses in the version of the model as first used in the study. Under certain conditions it underestimated the number of fighters scrambled against enemy raids, and hence underestimated the number of interceptions achieved). While the level of detail in the model made it operationally acceptable, the model results were not analytically understandable, so that robust conclusions of practical value could not be guaranteed. Consequently it was decided to examine raid indirect routing analytically, concentrating on the essential features of the problem. This approach was successful, in that the interactions between the various elements of the problem became evident, and the results of the study were more meaningful. The broad features of the problem are identified with the aid of the analytical model, while the significance of more involved or interactive elements can be examined with the simulation model. Part 3 of the thesis presents an extended and comprehensive version of this analytical model.

In the light of the study of raid indirect routing, the balance between a mathematical approach and the use of a simulation model was better planned for the analysis of the effects of sensor information errors. An analytical model was constructed in order to appreciate the complexity of the interactions within the problem. The model concentrates on determining the maximum acceptable error in the estimate of the raid track such that a fighter can still intercept the raid, as a function of its 'warning distance'. This maximum acceptable error is calculated for an arbitrary delay before fighter take-off, and so the method can be applied to fighters scrambled at regular intervals after the warning is received.

Only after the development of the mathematical model was the simulation model used in the study. The simulation includes factors which are of operational significance but are difficult to represent analytically. It also calculates the fighter intercept effectiveness in a form which is difficult and time-consuming using the analytical model alone - viz. the number of interceptions achieved (out of a total of 20 fighters scrambled), as a function of the error in the estimate of the raid track, for a fixed warning distance. A detailed presentation of this analytical model, together with a summary of the use of the fighter model, is given in Part 4 of the thesis.

9. THE DEVELOPMENT OF MATHEMATICAL MODELS OF RAID INDIRECT ROUTING AND SENSOR INFORMATION ERRORS

In this Chapter we outline the mathematical models used to examine the effects of raid indirect routing and sensor information errors on fighter intercept capability. Many of the assumptions underlying the models are common to the studies of both raid indirect routing and sensor information errors. We outline these assumptions first before describing the aspects peculiar to each study.

The models consider a single fighter base defending a single offset target against attack from a concentrated (point) raid. The co-ordinate system utilised in the geometric analyses could be derived, for example, as in Figure 1. This illustrates the principal numerical inputs in the examples chosen in Parts 3 and 4. The fighter base providing the fighter defences represents Coningsby (at the origin of co-ordinates), while the y-axis represents a 'target axis'. Fighter interceptions are regarded as successful only if they are achieved before the raid penetrates the target axis, with attacks coming from the hemisphere $x > 0$. This introduces a form of area defence and eliminates the need to define an expected target system, or to specify which fighter bases are likely to defend particular targets. With this convention we may take the raid target to be the intersection of the raid track with the target axis. In the numerical examples considered later we take this point to be half-way between the fighter base and its nearest neighbour along the target axis, i.e. Leuchars (see Figure 1). The distance perpendicular to the target axis at which detection by GC sensors first occurs is called the warning distance. A number of the values of warning distance considered in the examples are sufficiently small so as to correspond to low-level attacks, at high subsonic speed. It can be seen from the assumptions described here, namely:

(i) single fighter base;

(ii) point raid;

(iii) 'area defence' of a target axis;

(iv) maximum target offset along this axis;

(v) low-level attacks;

that the scenarios considered in the numerical examples are in a number of respects pessimistic for the fighter defences. Favourable conclusions concerning the viability of the fighter defences which can be drawn under these circumstances are then likely to hold also under more favourable conditions for fighter operations.

Following initial detection of the raid there is assumed to be a fixed delay for sensor and GC processing and transmission of the information, followed by a reaction delay representing the readiness level of the fighters. It is further assumed that, during these delays, the raid is identified as hostile and approaching (i.e. does not consist of stand-off jammers, patrolling outside defended airspace and emitting noise or deception jamming in support of the attacking raids), and is in sufficient strength to warrant fighter scramble. There is also a delay between each fighter scrambled for take-off. No limitation is assumed in the analysis on the number of fighters at the base, although in the numerical examples presented there are taken to be at most twenty.

Because of the underlying assumption that raid penetration is carried out at low-level, which places severe constraints on the upper speed limits of both the enemy raids and the fighters scrambled from ground alert, the fighter and raid profiles during the period in which raids are vulnerable to fighter attack are approximated by constant speed. Interception is represented by 'collision', the coincidence of the fighter and the raid on their respective tracks, with no allowance made for offsets for forward or rear engagements, or for re-attacks.

Indirect Routing

The raid is assumed to fly one feint leg, at the end of which it heads directly for its target. The position, speed and course of the raid at first detection, and at its track-change, are assumed correctly estimated by Ground Control. The raid track-change point is assumed to be pre-planned by the attacker, i.e. as opposed to evasive manoeuvre in direct response to fighter attack.

The maximum allowable fighter take-off delay for successful interception is used as a measure of fighter effectiveness. In the numerical examples, results are also presented in terms of the number of interceptions achieved before the raid reaches its target.

Four scramble and control policies are examined, as described below. The analysis is simplified considerably if interceptions are categorised into three distinct types. The effects of different raid feint tracks and warning distances on the number of interceptions achieved of each type may then be studied for the different scramble and control policies.

Scramble Policies

I Scramble is continued as long as interception is expected to occur before the target axis.

II Scramble is continued as long as fighters possess a geometrically feasible interception course.

III The feint leg is ignored and fighters fly along the target axis to a patrol position located directly over the target.

IV Scramble is continued as long as fighters can expect to intercept within a specified time (or distance) from the fighter base.

Categories of Interception

(i) Interception on the feint leg.

(ii) Interception on the second leg of the track by fighters re-allocated from expected interceptions on the feint leg, as a result of the raid track change.

(iii) Interception on the second track-leg by fighters scrambled to intercept the second track-leg. This comprises all interceptions under scramble policy III, while under the other policies it corresponds to fighters scrambled to intercept only after detection of the track-change by the warning sensors.

Because of the assumed accuracy and timeliness of the GC information, the fighters' radar detection performance may be ignored in this initial analysis. Thus it is assumed that, if an airborne fighter does not detect the raid track-change, it is nevertheless informed by Ground Control, after an appropriate processing delay, of its new interception course against the raid on its second track-leg. Further, it is assumed that the reaction delay to the track-change by fighters which are operating on GC information is equal to the reaction delay to the track-change by fighters which have already detected the raid and are operating autonomously.

The raid and fighter geometry is illustrated in Figure 2. The 'target' is at the point T with co-ordinates $(0, y_T)$. Each feint track has an associated feint angle ψ, i.e. the angle between the x-axis and the feint track. Given the feint angle ψ, the feint track is fully defined by specifying its point of intersection F with the target axis, or the distance x_T from the feint track to the target, measured perpendicular to the target axis. The track-change angle θ is the angle between the feint track and the second track-leg. Six configurations of the feint track and the second track-leg are considered in Part 3, and Figure 2 illustrates one of these, in which the raid feints north of the target and $\psi > 0$. This case represents the geography underlying the numerical examples discussed later.

The raid travels at a constant speed U and is first detected at the point S, with co-ordinates (x_o, y_o), where x_o is the warning distance. The raid changes track at the point C, which is defined either by the track-change angle θ or the feint time t_s, which is the time spent on the feint track after initial detection. Following this initial detection there is taken to be a fixed delay D_p for Ground Control processing and transmission of the sensor information, followed by a further reaction delay D_r representing the readiness level of the fighters. There is also a delay D_s between each fighter take-off, after which fighters fly with constant speed V.

If the first fighter scrambled also suffers the delay D_s then the total delay before take-off, D, for the I-th fighter scrambled is given by:

$$D = (D_p + D_r) + I.D_s \quad (I = 1, 2, 3 \ldots). \qquad (1)$$

Finally, it is assumed that, whether or not a fighter is capable of detecting the raid track-change on its own radar, it adopts if practicable its new interception course against the second track-leg after a delay D_c with respect to the time of the track-change.

Examples

The analysis of the number of interceptions achieved in each category, under the various scramble policies, for different feint tracks and warning distances, is presented in Appendices A and B of Part 3. Here we describe simple examples which are of interest and practical significance. In these examples the geographical scenario is shown in Figure 2. Fighters are assumed to fly at the same speed as the raid (V = U) and are assumed able to respond immediately to the track-change (D_c = 0), which is taken to be at right angles to the feint track ($\theta = \pi/2$). It is easily seen that equation (1) relates the maximum allowable take-off delay for successful interception to the corresponding number of interceptions, so the measure of effectiveness can be taken as the number of interceptions achieved before the raid reaches its target. These examples are analysed in detail in Part 3, for a range of numerical inputs. The outcome of this analysis is then discussed, and conclusions drawn as to the conditions under which indirect routing is beneficial to a raid, in terms of reducing the number of successul interceptions.

Sensor Information Errors

Errors in the initial Ground Control estimates of raid position, speed and heading at initial detection are considered. A pessimistic case for the fighters is examined, in which the initial erroneous GC estimate of the raid track is not updated, with presumably more accurate information, as the raid continues its approach. Hence fighters must themselves detect the raid before they correct their courses. This provides a 'worst-case' background for the study. In the analysis of sensor information errors it is assumed that raids do not engage in indirect routing but fly directly towards their targets. The measure of fighter effectiveness is the maximum acceptable error in the estimated raid track which still allows a fighter to intercept, expressed as a function of the timeliness of the warning provided by GC. (The analysis is then extended, by use of the fighter model, to give values for mean fighter intercept effectiveness for normally distributed errors with zero mean). Expressions are first derived for the maximum acceptable error under which fighters are scrambled (for scramble policies I, II, and IV, as defined in the study of raid indirect routing). The maximum acceptable errors that still enable a scrambled fighter to achieve radar detection of the raid are then determined. Finally, if a fighter detects the raid, the feasibility of it correcting its course and intercepting the raid is determined; if an interception is possible then the final interception point is calculated.

Fighters are assumed to have deterministic radar detection capability, defined by the radar range and angle of scan. Within the sector of the circle defined by these parameters a fighter detects with 100% probability, while it has zero probability of detection outside this sector. (This deterministic radar range parameter can be derived from a graph showing radar detection probability as a function of range, under given operating conditions).

Table 1 and Figure 3 summarise the raid parameters considered in the analysis. The x- and y-coordinates of the raid at first detection are termed the warning distance and offset respectively. The raid is believed to be first detected at the point S_f with co-ordinates (x_f, y_f), estimated speed U_f and estimated heading ψ_f; it is in fact first detected at the point S_t with co-ordinates (x_t, y_t), speed U_t and heading ψ_t. Given the scramble time and the initial fighter heading (determined by the GC estimate of the raid track), three parameters determine the fighters' capabilities - speed V (assumed constant), radar range R and angle of scan $\pm\alpha$.

Examples

The numerical examples presented in Part 4 refer to a raid which is flying at the same constant speed as the fighter and on a direct track to the target which is perpendicular to the target axis, so that

$$V = U_t$$

and

$$\psi_t = 0.$$

In the analysis of the various conditions for scramble, fighter detection of the raid, feasible interception and successful interception (i.e. before the raid reaches its target), errors in estimating the initial raid coordinates, speed and track are studied independently. A full discussion of these examples is given in Part 4.

10. SUMMARY AND CONCLUSIONS

In this thesis we have described the construction of a simulation model of fighter operations. This is an event-based simulation, set at the level of an air defence sector, which can run in either deterministic or stochastic mode. It concentrates on the representation of fighter radar, ECM and command and control. The role of the fighter model in a hierarchy of defence modelling techniques is illustrated.

The fighter model is applied to two operational problems in air defence, namely, the effects of raid indirect routing and sensor information errors on fighter intercept capability. Several deficiencies became apparent in the application of a detailed simulation model to these problems without additional mathematical analysis (cf. Lord (1978)):

- the model did not give an early enough quantitative appreciation of the interplay between the important variables;
- insufficient importance was paid to the wide range of numerical uncertainty in practically all the input data;
- there was an over-striving for depth of 'realistic' detail in order to achieve operational credibility and acceptance.

Mathematical models of both these aspects of fighter operations were constructed. These models concentrate on a deterministic treatment of the interactions between the important elements in the problems. This 'dual' approach of combined mathematical and simulation modelling was successful, in that the study results were both analytically understandable and operationally acceptable. The use of the simulation model was better planned, to include operational factors which were difficult to represent analytically, or to present information in a form which would have been impractical or time-consuming using analytical models alone.

OPERATIONAL CONCLUSIONS

Raid Indirect Routing

Ample warning distance tends to negate any possible advantage to the raid of indirect routing. For a target offset by about 100 nm from the fighter base, and for warning distances between 200 nm and 300 nm, the fighter intercept capability depends quite strongly on the scramble policy. There exists a range of raid feint angles which can drastically reduce the number of possible interceptions if a conservative scramble policy is adopted. By being prepared to scramble against an identified raid without close regard to the estimated position of interception the defence can nullify the possible deleterious effects of well-planned indirect routing.

It should be noted that, at the level at which this analysis is set, the possible disadvantages of such scramble policies are not considered here. These include, for example, the situation in which fighters are out of action at base for refuelling after an 'optimistic' scramble when further raids penetrate their specified area of cover.

Sensor Information Errors

If a reasonable warning distance can be achieved realistic errors in estimating the raid parameters do not degrade the fighter intercept capability, provided the range at which the fighters detect is not itself seriously degraded. The warning distance necessary depends on the number of fighters required to intercept the raid. To achieve 20 interceptions from a single fighter base against a raid on a perpendicular track to a target offset by about 100nm, errors in estimated offset distance and raid heading appear to be potentially the most serious. The assumed ability of the fighter radar to maintain its detection capability (by bearings-only analysis against self-screening jamming), together with the fighters' assumed ability to respond quickly to a detection, contribute significantly to their ability to tolerate GC sensor information errors.

The results presented go some way towards a classification of the more effective procedures which the defence might adopt in the face of deceptive raid tactics and degraded sensor performance. They indicate the value of ample warning distance to the fighter defences, together with early resolution of raid size in order that sufficient fighters may be scrambled quickly, albeit not necessarily with well-defined interception courses. However, the numerical examples presented are in no way definitive; they are selected to display certain facets of the problems although many interactions between variables remain unexplored.

TABLE 1

PARAMETERS INVOLVED IN THE ANALYSIS

RAID PARAMETERS AT INITIAL DETECTION

	Actual	Estimated by Ground Control (GC)
x-coordinate(warning distance)	x_t	x_f
y-coordinate (offset)	y_t	y_f
range from fighter base	r_t	r_f
bearing " " "	θ_t	θ_f
heading	ψ_t	ψ_f
speed (constant)	U_t	U_f

FIGHTER PARAMETERS

speed (constant)	V
delay before take-off	D
radar range	R
radar angle of scan	$\pm\alpha$

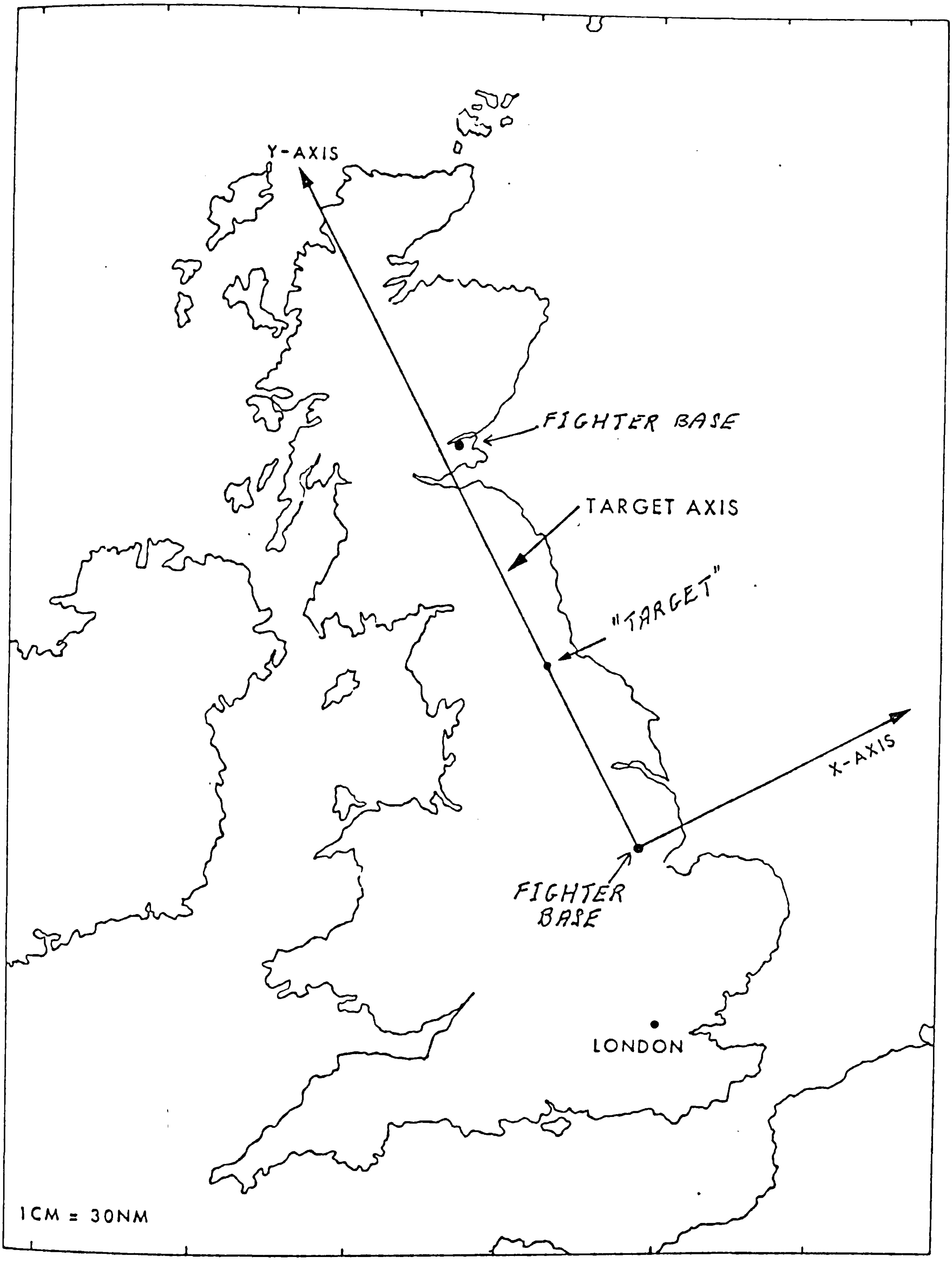
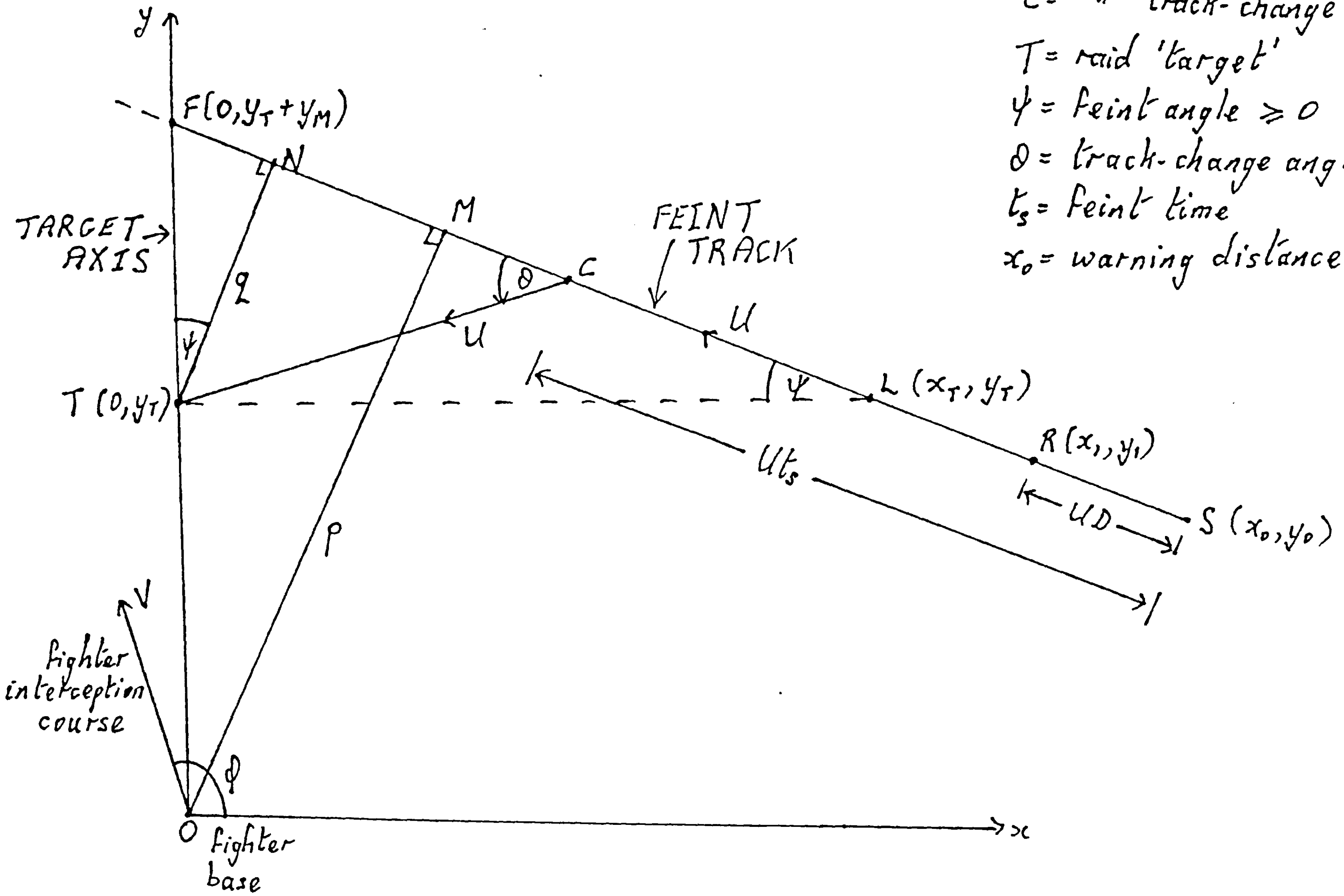


FIGURE 1: TARGET AXIS AND CO-ORDINATE SYSTEM



S = raid position when first detected

R = " " " fighter scramble:

C = " track-change point

T = raid 'target'

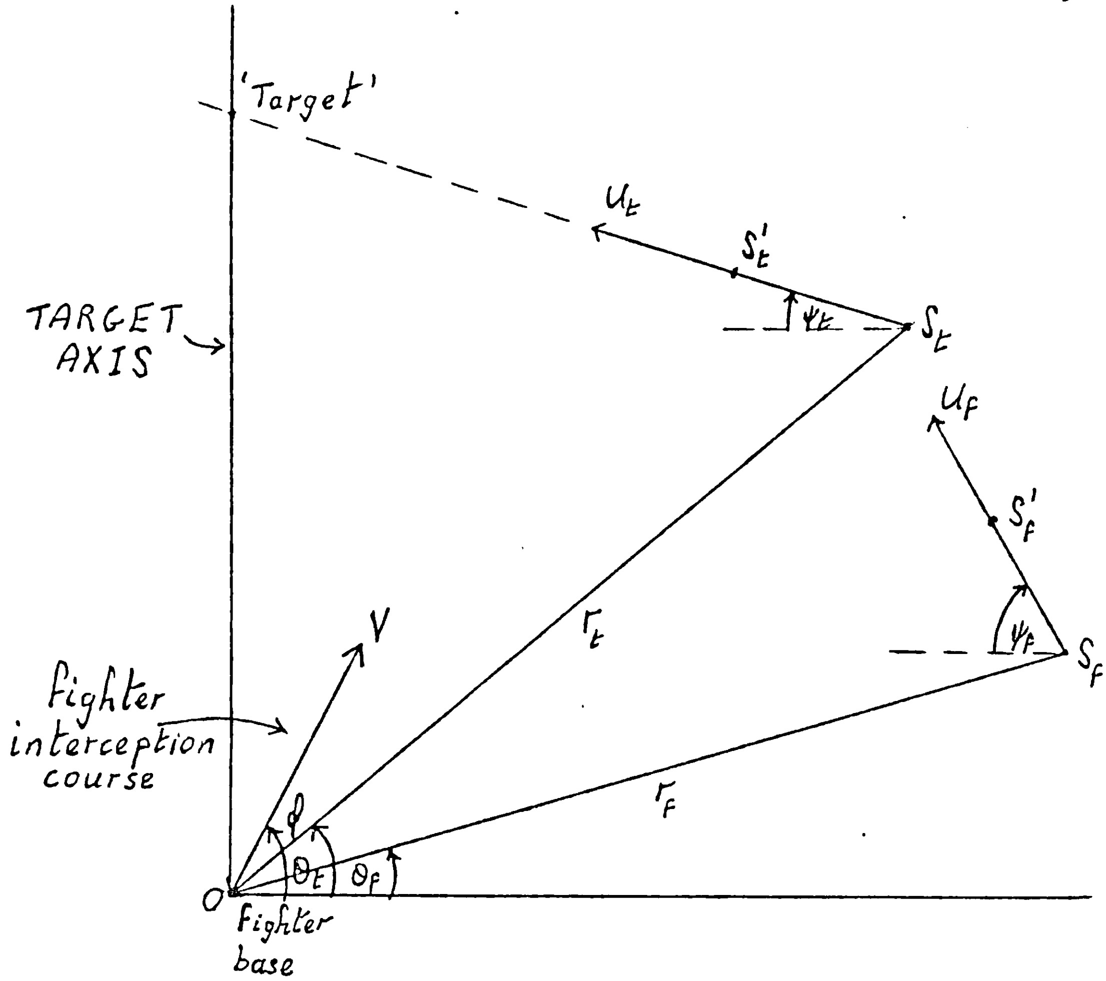
ψ = feint angle ≥ 0

θ = track-change angle ≥ 0

t_s = feint time

x_0 = warning distance

Figure 2. Raid and Fighter Geometry (case (1))



S_t = actual raid position at first detection by GC

S_f = estimated " " " " " " "

S_t' = actual " " " fighter scramble

S_f' = estimated " " " " "

Figure 3 . Fighter and Raid Parameters

PART 2

THE FIGHTER MODEL

CONTENTS OF PART 2 - THE FIGHTER MODEL

PART 2.1

GUIDE TO THE FIGHTER MODEL

INTRODUCTION

1. Three complementary models representing the major components of air defence operations - Air Defence Ground Environment (ADGE), fighters and surface-to-air missiles (SAMs) have been developed at DOAE. This paper describes the Fighter Model.

2. The aim is to give a general description of the model and to suggest ways in which it could be further developed in Part 2.1, while Part 2.2 is a detailed description of the model structure. The introductory paragraphs of Part 2.1 describe the general scenario to be represented and the model structure, and list the event types included in the simulation. These are followed by sections covering enemy raids, fighter activities, command and control, threat matching and fighter allocation, a summary of the main sequence of events for fighters, AI radar and jamming, and fighter response to AI detections. Some technical details on the event processing of fighters are then described, while the final sections of Part 2.1 summarise the most important assumptions made in the model and outline possible extensions which could increase its scope and applicability.

3. Part 2.2 of the paper explains the processing of each subroutine in the model, taken in alphabetical order. Annex A is a glossary of the variable names which occur in the model, while Annex B is an example of a simple test data file for use with the model. This gives some idea both of the scenarios envisaged and of the sort of data which will need to be collected for any study involving the model.

4. The model represents operations at the level of an air defence sector; at this level substantial simplification is essential. In particular, it was considered that the detailed representation of missile flight paths, fusing and lethality and the representation of dog-fights could not be included in a fighter model at this level without making it impossibly complex. The model concentrates on the representation of AI radar, ECM and command and control, whereas the representation of fighter aerodynamics is highly simplified. Performance data on acceleration and fuel consumption rates in various flight profiles must be specified, for each fighter type, in the input data; the model does not calculate these parameters itself.

MODEL SCENARIO

5. The scenario envisaged for the model concerns the incursion into friendly airspace - for example, the UK Air Defence Region (UKADR) - of a number of subsonic or supersonic bomber raids, with possible support from specialist escort jammers and stand-off jammers. The duration of the incursion is determined by the time taken for the bombers to fly from the limits of early warning cover to their targets and back again. Enemy aircraft will be

vulnerable to attack throughout this period.

6. The model can represent alternative tactics on both the attacking and defending sides, and insofar as tactics are explicitly represented, details are given at the appropriate places in the paper. It is not the intention of the model, however, to define the tactical behaviour of a force, but rather to leave tactics as much as possible a function of model input data. It is important to note that a long term air defence strategy is not represented, and it is not currently planned to incorporate such a campaign analysis facility in the model. In particular the recovery, turn-round and serviceability of fighters is not directly represented. Also the allocation policy is concerned only with matching the immediate threats, rather than representing some overall strategy.

7. Regions of airspace prohibited to fighters (eg Missile Engagement Zones) are represented in the model by series of straight-line boundaries which fighters may not cross.

MODEL STRUCTURE

8. The model is an 'event-based' simulation, any significant occurrence to any entity in the model being called an event for that entity. For example, events for bomber raids would include track-change points; events for fighters include take-off, change of acceleration or fuel-consumption rate, and detection of target. As a final example, the total radar and ECM picture is updated throughout a model run, the time-interval between occurrences of this process corresponding roughly to the scan-time of the fighters' radars. The event which triggers this process is called the radar-scan event.

9. At an occurrence of any event all possible consequences of that event are considered and, in particular, any new events which are generated must be set up. Thus if there is little activity at any stage in a model run few events will occur and hence computer time will not be wasted in needless updating. Conversely, if there is a great deal of activity in a short period of time, each event will be considered and acted upon separately, in chronological order. Thus the heart of the program is simply a list of all events and the data associated with each event - the time at which it is to occur, the type of the event and the entity to which it occurs. As each event occurs and its consequences are examined this list is amended as necessary, with new events being inserted and cancelled events deleted.

Mode of Operation

10. Two versions of the model have been written, a stochastic ('Monte Carlo') version and a deterministic ('expected value') version. In the Monte Carlo model, whenever the value of a random variable is required, it is derived from a suitable distribution using a random number generator. The results of the simulation are also random variables, and to obtain reliable information concerning their average values the program must be run many times with independent sets of random numbers. In the deterministic model, on the other hand, the use of each random variable is replaced by a process intended to represent the 'expected result' of using

Again because of the possible Ground Control (GC) reaction delay, this is done via a subsidiary event, type 20.

Event type 3 corresponds to the first detection of a raid by Ground Control. It is generated when the raid crosses warning line 1 (para 29). A specified number of fighters is scrambled to a holding CAP on a least-time-to-CAP basis. After take-off these fighters search for enemy raids on their AI radars.

Event types 4 and 13 correspond to the acquisition of track information on a raid by Ground Control. Event type 4 is generated when the raid crosses warning line 2 (para 29) and event type 13 when a fighter achieves burnthrough on the raid. The processing associated with these events is very similar. Fighters already attacking the raid are given new interception courses. A specified number of fighters is required to attack the raid once track information is available; an attempt is made to meet any shortfall, preferably from fighters cruising to or on CAP, or else from unallocated fighters on alert at the fighter bases.

Event type 5 corresponds to the acquisition by Ground Control of accurate information on raid strength. It is generated either when a raid crosses warning line 3 (para 29) or when a fighter is sufficiently close to the raid to resolve it on its AI radar. The input data contains a parameter which specifies the ratio of fighters: enemy aircraft which GC attempts to achieve when the strength of a raid is known.

Event type 6 is the most complex event in the model. It corresponds to the updating, at a specified frequency, of the complete radar and ECM picture for each fighter and raid. In particular it determines, for each fighter and raid, the raid signal strength at the fighter, the total jamming experienced by the fighter when it is illuminating the raid and the fighter's background noise and clutter levels. Knowing these values the fighter's response to the raid is then determined.

Event type 7 is a fighter take-off. The fighter's first change of profile-point (event type 8) is set up at take-off.

Event type 8 corresponds to a fighter reaching a profile-point, at which its acceleration or fuel consumption rate changes; each event type 8 generates its successor.

Event type 9 Fighter arrives on CAP and

Event type 10 Fighter reaches a CAP point.

A distinction between these two events is only drawn in order to make the arrival of a fighter on CAP distinct from its ensuing progression through the CAP points.

Event type 11 Fighter returns to base. This event is generated if a fighter, unable to continue a planned interception, cannot reach any of the pre-defined CAP patterns. While flying back to base the fighter may detect and attack enemy raids on its AI radar.

Event type 12 Fighter reaches its expected interception point. The measure of fighter effectiveness adopted in the model may be event type 12 (collision) or event type 16 (missile splash); this is controlled by input data. With either measure, if a fighter's interception course is so much in error that it reaches its expected interception point without even detecting its target, the planned interception is abandoned. If the fighter has detected its target and the measure of effectiveness is event type 16, this event is ignored while the fighter continues its attack. Otherwise, if collision suffices as a measure of fighter effectiveness, it is unnecessary to model reattack sequences, so that (after printing interception information and collecting statistical data) no further processing is required at this event.

Event type 13 GC acquires track information from a fighter. The processing associated with this event is very similar to that of event type 4; the two event types enable the source of track information on a raid to be identified.

Event type 14 recalculates a fighter's interception course, after a track-change by its target.

Event type 15 spare.

Event type 16 missile splash. The missile kill probability, P_k, is calculated; this depends upon the missile type, the ECM conditions and the hemisphere of the attack. In stochastic mode a random variable determines if a kill has been achieved, while in deterministic mode fractional numbers of enemy aircraft are allowed and exactly P_k aircraft are eliminated. The fighter then enters a reattack sequence to choose its next target.

Event type 17 missile launch check. This event tests at discrete intervals the range from a fighter to its target to determine if it is within missile launch range. It it is not the next event type 17 for this fighter is generated, while if it is within range a missile is launched and the missile splash event (type 16) is generated.

Event type 18 Fighter released after a turn. The simplification is currently adopted that the spatial movement of a fighter during a turn between attacks in a reattack sequence may be neglected, ie its x- and y- coordinates remain frozen. This event releases a fighter at the end of such a turn in order that it may continue on its next planned interception.

Event type 19 spare.

Event type 20 is generated by a raid track-change (event type 2). It tests if there is any shortfall in the number of fighters attacking the raid after its track-change. If so it attempts to meet this shortfall, preferably from fighters cruising to or on CAP, or else from unallocated fighters on alert at the fighter bases.

THE REPRESENTATION OF ENEMY AIRCRAFT

12. The model is designed to represent attack by a number of groups of enemy aircraft. The size of each group can be varied - anything from a single aircraft to a large rectangular array of aircraft. The flight path of each group is specified by a series of line segments, each segment being flown at constant heading, speed and height. Bomber weapon release occurs at a pre-specified track change point. Raids and fighters are interactive inasmuch as the jamming output of a raid may depend upon the number, type and strength of the radar detections of that raid by the fighters. Detections do not affect the raid tracks, ie fully interactive raid behaviour, incorporating evasive manoeuvres, is not modelled.

13. Enemy aircraft are vulnerable to attack whenever they are outside designated SAM engagement zones, and they remain vulnerable even when returning eastwards having completed their missions. The event sequence of a raid terminates if all the aircraft within the raid are killed.

14. Two types of aircraft may be present within a raid; bombers (which may be self-screening jammers) and specialist escort jammers. Fighter escorts are not modelled. It is not possible to represent continuously the 3-D position of every fighter and every enemy aircraft in the model - this would make the program far too inefficient. The representation of the detailed structure of enemy raids is therefore varied according to the requirements at the time. A raid is normally considered as concentrated at a point, for example when interception courses are calculated, and the jamming output of a raid is normally considered as a point source of jamming. Providing the group of aircraft described in the model as a 'raid' does not extend over too great a distance, this is a reasonable approximation. The structure of a raid is considered in detail only in determining:

a. Its radar cross-section, in terms of the cross-sections of its constituent bomber-type and specialist jammer-type.

b. A resolution factor, viz the ratio of the width of the raid (as perceived by a fighter) to the beamwidth of that fighter's radar when it illuminates the raid. It is a crude measure of the number of independent glimpses of a raid which a fighter acquires when its radar scans through it, and is used in determining the probability of detection of the raid by the fighter and the effective jamming power of the raid.

c. The number of aircraft in the raid, referred to in this paper as the raid size; the point at which a fighter determines raid size of course depends upon the aircraft spacing within the raid. (It is assumed that raid size is determined accurately by actually resolving the raid on the fighter's radar rather than by crude estimation from the total radar echoing area).

d. The next target within a raid for a fighter in a reattack sequence.

15. It is worth mentioning here a point of notation. The standard grouping of enemy aircraft in the model is a rectangular array of aircraft in m rows and n columns, where m or n or both may equal 1 (see figure 0.1). This of course is an idealised structure, and may not at all approximate to the raid structure in the particular scenarios under consideration. Nevertheless, since any number of such groups may be defined in the model, clearly a raid of arbitrary size and shape may be specified by an appropriate combination of such groups. For the sake of simplicity only, the single group of aircraft illustrated in figure 0.1 is taken in the rest of this paper to be a typical raid, while the columns of such a raid are referred to as <u>flights</u>. So far it has not been necessary to represent individual enemy aircraft explicitly. The numbers of each type of aircraft within each flight of a raid must be specified in the input data, but the positions of individual aircraft within a flight are not specified. After a successful missile-firing the type of target killed is decided by sampling from a suitable distribution.

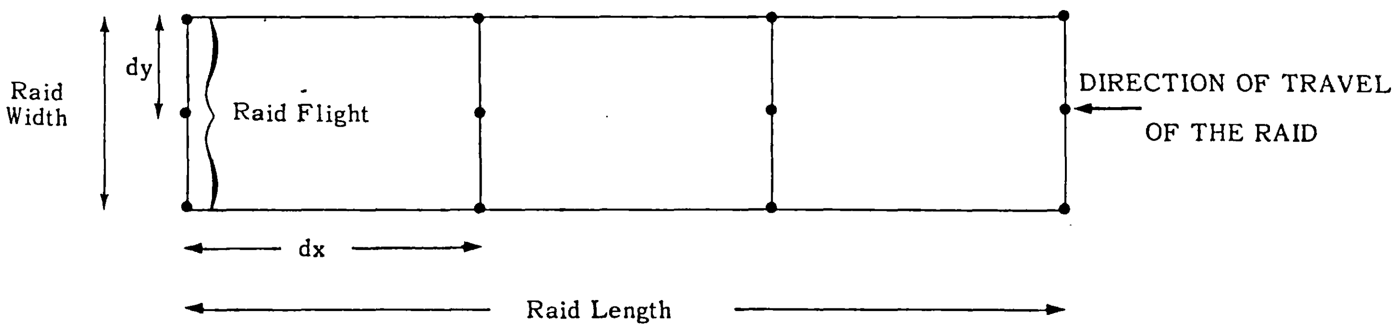


FIGURE 0.1. TYPICAL STRUCTURE OF AN ENEMY RAID

Notes:

This is an example of a group of 12 aircraft; the separations dx, dy are specified in the input data.

THE REPRESENTATION OF FIGHTERS

16. When a model run starts, fighters will be either already airborne in loiter patterns, or waiting at air bases in queues to be scrambled. The control routine, depending on available information, then allocates fighters either to intercept enemy raids or to particular CAPs, whence the fighters are free to make their own detections and interceptions using AI radar. Interception courses are more or less accurate depending on the degree of control available and the accuracy of track information. When interceptions finally occur, fighters attack enemy aircraft with either radar or infra-red homing missiles. Fighters, having completed first attacks, may reattack further aircraft either in the same raid or in other raids. The following paragraphs each describe a specific aspect of fighter operations.

Flight Paths

17. Flight paths are represented by storing for each aircraft type a number of flight profiles in the form of velocity: time points and fuel-consumption rate: time points, approximating the actual performance of the aircraft. Different profiles can be envisaged for different fighter altitudes, different enemy altitudes and different fighter store configurations (drop-tanks in particular). These profiles will be specified as input data using the available estimates of fighter performance.

18. Each fighter is only permitted to intercept and engage an enemy aircraft if this leaves it with enough fuel to return to a base and since the fuel consumed during combat is unpredictable, threshold fuel levels are set beyond which certain activities are prohibited. In-flight refuelling of the fighters is not included in the current version of the model.

19. The model incorporates the effects of altitude only implicitly, via data. Two principle aspects of fighter operations are importantly affected by altitude considerations:

a. A climb usually requires either more time or more fuel (usually considerably more of the latter) than a descent or level flight over the same horizontal distance. This aspect of altitude can be included in the velocity: time and fuel: time curves used for assessing flight profiles.

b. The pulse and pulse doppler clutter calculations involve a degree of approximation at present. The clutter experienced by a fighter depends on its altitude and orientation in the vertical plane and the model does not regularly update these parameters.

Air-to-Air Missiles

20. The fighter model represents both radar and infra-red homing missiles. For each missile a representation of the maximum and minimum launch envelopes is stored, together with the corresponding time-of-flight, as a function of target aspect angle. The effectiveness of a missile after launch is represented as simply as possible by a series of kill probabilities,

which are a function of two factors:

(i) Whether the attack is essentially forward hemisphere or rear hemisphere.

(ii) The ECM conditions prevailing, at time of launch for infra-red missiles and atmissile-splash for radar missiles. (The difference is because radar missiles need to be guided during their flight). Three such conditions are modelled:

a. Clear

b. Home-on-Jam (HOJ). In this mode the fighter, having achieved burnthrough and so adopted an autonomous interception course, is denied current range information. In the model the fighter is assumed to continue on its interception course and to launch a missile when within the launch envelope. Any degradation in missile effectiveness when it is launched in this reversionary mode is subsumed into the final missile lethality.

c. Angle-on-Jam (AOJ). This mode gives the lowest kill probability, it being the reversionary mode when the fighter has no range information.

It is left to the input data on missile launch envelope and kill probability to reflect the fact that IR homing missiles are normally only fired successfully in a rear hemisphere attack, while radar homing missiles can be fired from behind or ahead of the enemy, the latter being preferred. The effect of an altitude difference between a fighter and its target on missile kill probability, in a snap-up or snap-down attack, is not represented.

Air bases

21. During the time-period envisaged for the fighter model scenarios, it is not expected that turn-round of returning fighters at air bases will contribute significantly to the air defence effort. Hence air bases are represented very simply as queues of aircraft able to take off at regular time intervals when required, and aircraft returning to base are subsequently ignored. Apart from its position and the numberand availability of its aircraft, an air base has no other significance in the model.

CAPs

22. CAPs are represented in the model by a series of points, and fighters fly straight lines between these points in a specified order; thus CAPs of arbitrary shape and orientation may be defined. Fighters on CAP cruise at an input height, speed and fuel-consumption rate. All aircraft on CAP utilise their radars to detect enemy targets; fighters may also be tasked to close-controlled interceptions from CAP. Hence CAPs are treated both as a means of detection and as a source of fighters.

COMMAND AND CONTROL

23. The aspects of command and control represented in the fighter model are those activities of an SOC (Sector Operations Centre) or a CRC (Control and Reporting Centre) which particularly affect fighters. This is not done by modelling any components of the control centre, but by concentrating on the interactions between the control centre and the fighter force.

24. Two general principles characterise the command and control routines and while they are not thought to be unrealistic, they do illustrate the somewhat ideal approach used:

a. The decision-making is effectively short term, each decision being a more or less immediate responseto a new item of incoming information. This approach is quite deliberate, since it simplifies the logic and matches the fairly short term scenario featured in the model, but it does omit any longer term considerations such as planning for special enemy raids expected in the future, and at no stage does it analyse the deployment of the whole air defence fighter force.

b. The other principle of note is that the allocation of fighters to specific targets is not changed unless interceptions become impossible.

25. There are four control modes currently represented in the model - close control, broadcast control, data link control and autonomous control. These control modes are not represented directly, but implicitly via their effects on fighter operations; in particular, the control mode of a fighter determines its communications delays and heading errors.

26. Whenever possible, a fighter will be close-controlled along its interception course. If there are insufficient intercept controller positions available it is directed by broadcast control, which is normally subject to increased error and delay. When a fighter has detected its target, it is assumed in the model to be released from its intercept controller, if any, and to operate autonomously.

27, Command and control is represented as an information-processing function, and the control of fighters can be broken down into three items:

a. Maintaining an up-to-date picture of the current situation, as regards both enemy threats and friendly aircraft, as a basis for decision-making. In addition, the future availability of air defence fighters, as regards crews, aircraft, bases, etc forms an important part of the total picture.

b. Deciding how best to meet the current threat and any anticipated future threat.

c. Implementing the decision of b., in the form of tasking individual aircraft. Details of the representation of these items in the model are given below, while the next section considers item b. in more general terms.

28. The structure of the control routine in the model is straightforward; each new piece of information received is analysed and then acted upon, after an appropriate time delay. Information either originates exogenously, to represent the acquisition of information by Air Defence Ground Environment (ADGE) radars, or else it is generated when some event occurs to some entity in the model; for example, fighters may report details of a kill or an enemy track-change back to the control centre. For simplicity, only a limited number of information categories are represented in the model; they are described in the following paragraphs.

29. A detailed representation of ground-based EW or AEW radar is not attempted. Instead, within the model information about enemy raids is generated, with appropriate errors and communications delays, either by the fighters themselves or when the raids cross a series of three 'warning lines'. Each warning line is represented by a series of line-segments, the positions of which are data inputs to the model. Data in this form can be generated by a model of the Air Defence Ground Environment, or it can simply be generated manually. Warning line 1 corresponds to first knowledge of the existence of the raid, line 2 to knowledge of its present position, velocity and altitude, and line 3 to knowledge of the raid size. The following paragraphs describe the processing carried out within the model when this information becomes available.

Initial Detection of an Enemy Raid

30. This is the most limited category of information concerning an enemy raid represented in the model, comprising knowledge of its existence, but with only very rough quantitative information. Initial detection occurs when a raid crosses warning line 1, which could correspond to, eg, the limit of UKADGE radar coverage. When a raid crosses this line a number of fighters (up to a maximum specified in the input data) are scrambled to a holding CAP. The fighters are chosen on a least-time-to-CAP basis, assuming that they fly to the CAP in a cruise profile. After take-off their own radars begin operating in the search for enemy aircraft. With a fast raid, of course, detections may occur even before the CAP is reached.

Raid Track Information

31. Information on raid position, speed and heading is generated when a raid crosses warning line 2. Delays and errors in the estimates of these parameters, representing degraded communications, can be specified in the input data. Note that this information may also be generated by any fighter which is tracking the raid on its radar. Errors in the estimates of the raid parameters are assumed to be generated from normal distributions with zero mean. Each fighter type has specified as input its air-ground communications delay together with the standard deviations of its error distributions. Regardless of where the information came from the model then finds and allocates a number of fighters to intercept

the raid. These are chosen on a least-time-to-intercept basis, assuming the appropriate interception profile has been selected; the number of fighters allocated is specified in the input data. The interception algorithm also takes into account fuel and out-of-bound limitations and heading errors.

32. A raid track change, after the generation of its initial track information as described above, also comes into this category. When this occurs in the model, every fighter allocated to the raid is tested to see if it can still intercept. (For example, a fighter may no longer be able to intercept because of inadequate fuel reserves, or because the new interception point is outside the permitted fighter area). If a fighter can still intercept, its new interception course is adopted, otherwise it is allocated to another raid, if possible, or else it cruises to a holding CAP. A new fighter is found to replace it in the attack on the original raid, again on a least-time-to-intercept basis.

Raid Size Information

33. Raid size information consists of a useful indication of the number of aircraft in the raid. It is generated either when a raid crosses warning line 3, or by the fighters themselves when the raid gets sufficiently close to be resolved on their AI radars. This information enables the number of fighters assigned to the raid to be changed to meet the desired fighter: attacker ratio (specified in the input data). Is is of course possible that a raid may never reach warning lines 2 or 3, if for example it flies at very low level in the absence of Airborne Early Warning (AEW), or if it is designed as a deliberate spoof, with limited penetration of defended air-space.

Information that a Fighter is 'Free"

34. Fighters become 'free' for further tasks in the model either when an enemy raid is completely destroyed, or when a planned interception is no longer feasible, perhaps because the enemy has changed track. A 'free' fighter is provisionally sent to a CAP, where it continues searching on its AI radar for new targets; it is also available for reallocation by Ground Control to attack another target.

Information that a Fighter has Detected its Target

35. This information is of significance because, in the model, it represents the assumption of intercept control by the fighter,possibly freeing an intercept position at the control centre for another close-controlled interception.

THREAT MATCHING AND ALLOCATION POLICY

36. One of the major objectives of a sector controller must be to maximise the number of enemy aircraft shot down within his sector. In addition he will have several subsidiary objectives, such as minimising his own losses and protecting key points from attack. It is however impossible to define in advance an 'optimum' strategy, guaranteed to meet

these objectives better than any other. In practice neither the exact nature of the enemy threat nor the time-horizon of interest can be known in advance, and variations in either of these factors can make major differences to the effectiveness of any given strategy.

37. The strategy of the controller cannot therefore be quantified by any formal algorithm, and in the model it takes the form of a set of flexible guidelines, as follows:

a. Share out the defence resources among enemy threats in proportion to their 'importance' (input data).

b. Intercept a given enemy raid with the nearest available fighters, to save time and fuel.

c. Maintain a reserve capability to deal with future contingencies.

d. Make aircraft deployments flexible (eg use CAPS as staging points).

e. Keep interceptions and recoveries flowing smoothly so that bottlenecks do not develop at bases.

38.Threat evaluation in the model is straightforward, the importance of a given enemy raid being measured by two parameters, namely its size and the time at which it is expected to enter a designated target area. This simplification ignores such problems as the different possible weapon-loads per aircraft in the raid, and the difficulty in determining the raid's target if it adopts a zig-zag flight path. The sector controller is assumed to possess accurate and up-to-date information on his own air defence resources - the positions, velocities, fuel and weapon states of the fighters,and also the situation regarding aircraft availability at air bases.

39. The allocation policy adopted in the model is then as follows. On first ascertaining a raid's track, generally before information is available on its size, a fixed number of fighters is allocated, on a least-time-to-intercept basis, to intercept the raid. Later, when raid size information becomes available, further fighters may be allocated until a specified fighter: enemy aircraft ratio is reached. In the current version of the model the time to the target area is ignored, on the grounds that the order in which the raids are first detected is a realistic proxy order of priority.

40. Note that the fundamental criterion used for choosing fighters to match threats is that of 'least-time-to-intercept'. This is not unreasonable, but may not always represent the 'best' policy. For example, while fuel consumption and limitations are taken into account, in some scenarios the desire to conserve fighter fuel rather than minimise bomber penetration may be the dominant factor. Nevertheless it was felt that enemy penetration distance should be the factor given prominence in the model logic.

FIGHTER MAIN EVENT SEQUENCE

41. Figure 0.2 shows very briefly the main sequence of events for fighters, excluding details of the AI radar processing which is described later. The following notes should help to explain the features of the flowchart:

(1) When receiving a new assignment a fighter may be on the ground at an air base, or already in the air. Any scramble time must therefore be allowed for in evaluating the new fighter course, but this point is not worth showing separately on the flowchart.

(2) A fighter will be assumed to fly to the nearest point of a CAP at cruise speed, before beginning to fly along the CAP line.

(3) In principle a fighter can receive new orders from ground control at any time, instructing it to change to some new task. In fact changing the tasking of a fighter under close control or autonomous control is avoided, unless the fighter's current mission becomes impossible (such as when its target is destroyed, or when it can no longer intercept due to a track change by its target).

(4) The time for which fuel lasts follows from the fuel consumption rate at the CAP cruise speed and the fuel threshold beyond which insufficient fuel is left to carry out an interception and return to base.

(5) For the purposes of this flowchart close control, broadcast control and control by another fighter via a data link are indistinguishable.

(6) In a correctly set-up interception AI detection should occur some time before the calculated interception point. It is necessary to impose a time limit on AI detection, however, to deal with the case in which the adopted interception course may be so much in error that target detection is not in fact possible.

(7) The action taken when a fighter fails to detect its target is to provisionally send it to CAP while it selects a new target, if possible, on the basis of its AI radar returns.

(8) The section of flowchart below this point, illustrating the fighter response to an AI detection, is expanded in more detail in figure 0.5.

(9) Having fired one missile (or two at once, possibly, this option being represented by suitable input data), the fighter pilot has the choice of either reattacking the same aircraft, if it is still alive, or of attacking some other aircraft in the raid. The action actually taken is a function of the structure and speed of the raid, and is expanded in the description of the reattack subroutines.

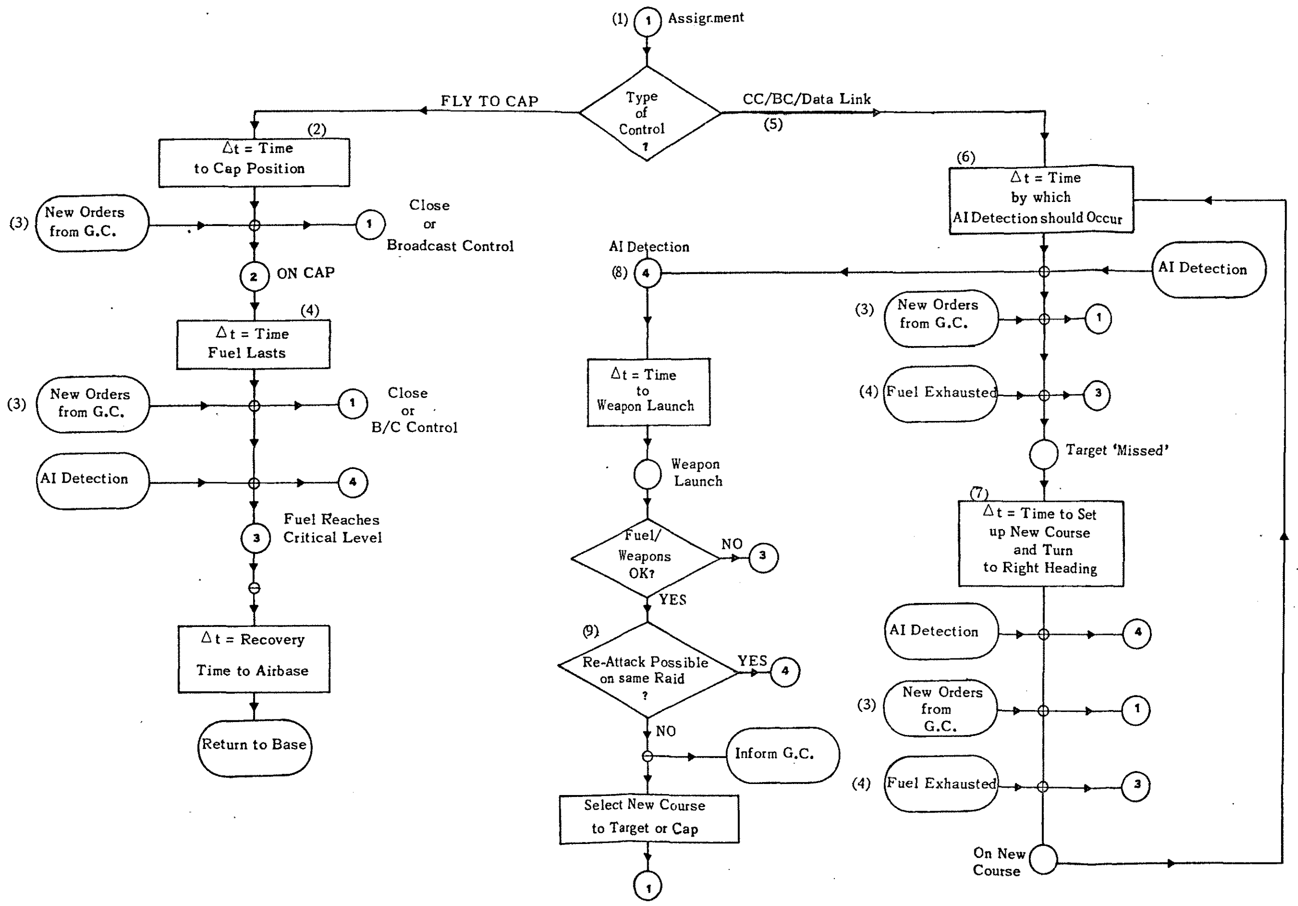


FIGURE 0.2. FIGHTER MAIN EVENT SEQUENCE (SEE MAIN TEXT FOR NOTES)

THE REPRESENTATION OF AI RADAR AND JAMMING

42. The section of the model dealing with radar and jamming uses as much computer runtime as the rest of the model put together. This is partly because a complex process is being modelled, (radar echoing areas change significantly with azimuth angle, requiring frequent signal strength reassessment, while each jammer can in principle respond to every radar transmitter) and partly because radar calculations are inherently time-consuming, requiring large numbers of floating-point multiplications. The design of the radar routines has therefore received particular attention, with the emphasis on simplification where possible.

43. The only type of radar explicitly represented in the model is the fighter AI radar, capable of operating in three distinct modes - scanning, tracking and lock-on. More advanced radars can in fact track targets while still in the scanning mode - 'track while scan'. Two types of jammer are represented, the stand-off jammer emitting continuous barrage jamming, and escort and self-screening jammers, which form part of each enemy raid and will generally respond actively to detected threat radar signals. Each escort or self-screening jammer may operate in one of three modes - it may jam continuously, like the stand-off jammers; it may jam only those radars which it detects continuously (ie radars which are sufficiently close to be detected at the sidelobe level, or which are locked-on to the jammer), or finally it may also be capable of jamming scanning radars, during their paint only.

44. The processing of radar and jammer behaviour is carried out in self-contained subroutines. The general structure of these routines is determined by the need first of all to evaluate the radar signal strength detected at each jammer, then to generate the response of each jammer, and finally to determine the performance of each radar in the presence of jamming and clutter, and hence determine the response of each fighter. There are three distinct logics involved in this process - jammer detection logic, jammer response logic and fighter response logic. Flowcharts illustrating the particular logics in the current version of the model are given in figures 0.3-0.5. The self-contained nature of these subroutines means that different jammer logics and pilot tactics can be substituted with relative ease.

45. The total radar and ECM picture is updated regularly during a run of the model, by the radar-scan event. The first task at each event is to update the positions of all airborne fighters and raids. This determines all ranges and angles which may be required later. The jamming power of each raid is then determined; depending on the nature and sensitivity of the jammers, various detection and response logics could be chosen. The cumulative effect of these jamming responses upon each fighter's detection of each raid is then calculated, together with the clutter, noise and target signal strengths. The signal-to-total noise ratio (SNR), jammer/noise ratio (J/N), signal/jammer ratio (S/J) and probability of detection can then be found, for each fighter-raid pair. Knowing these values, the response of each fighter can finally be determined.

46. Figure 0.3 is a flowchart of the radar and jammer routine, and the following notes enlarge upon some of its features.

(1) Between calls of the radar routine, the positions of all aircraft will in general change, altering ranges, look angles and aspect angles. This alters all radar and jammer strengths, both because of the changes in range and the effects of jammer and echoing-area polar diagrams. Changes in aircraft position and velocity also affect the magnitude of radar clutter returns.

(2) In the first section of the radar routine the term 'jammer' refers only to responsive jammers: stand-off jammers need not be considered since their jamming output is predetermined.

(3) 'Jammer detection logic' refers to the way in which incident radar signals are perceived by the jammer - either they are not detected at all, or they are continuously detected and jammed, or they may only be detected and jammed intermittently.

(4) 'Jammer response logic' refers to the way in which the bandwidth of the jammer output depends on the perceived incident radar signals.

(5) The second jammer loop <u>does</u> include stand-off jammers.

(6) The results of the radar and jamming calculations comprise values for radar signal strength S, perceived jammer strength J (in the direction of the same target) and radar clutter strength C. AI radars operate in pulse doppler mode; pulse clutter calculations can be incorporated in the program when the criteria used by the aircrew when selecting pulse/pulse doppler modes can be quantified. The radar noise power, N, is specified in the input data for each radar type. The signal-to-total noise ratio, SNR, is then given by:

$$SNR=S/(C+J+N)$$

The criteria for acquisition of range are simply comparison of SNR, J/N and S/J with the relevant threshold values (specified in the input data), as is the criterion for initial detection in the deterministic version of the model. In the Monte-Carlo version it is more appropriate to calculate a detection probability, which is then compared with a random number from a uniform distribution between 0 and 1.

(7) These calculations are performed for each fighter and at each occurrence of the radar-scan event. If a fighter is already allocated to a particular raid, they are performed for that raid only. If a fighter is not allocated to a target they are repeated for each raid, in the hope of finding it a suitable target.

(8) The response of each fighter can now be determined, as explained in paras 49-52.

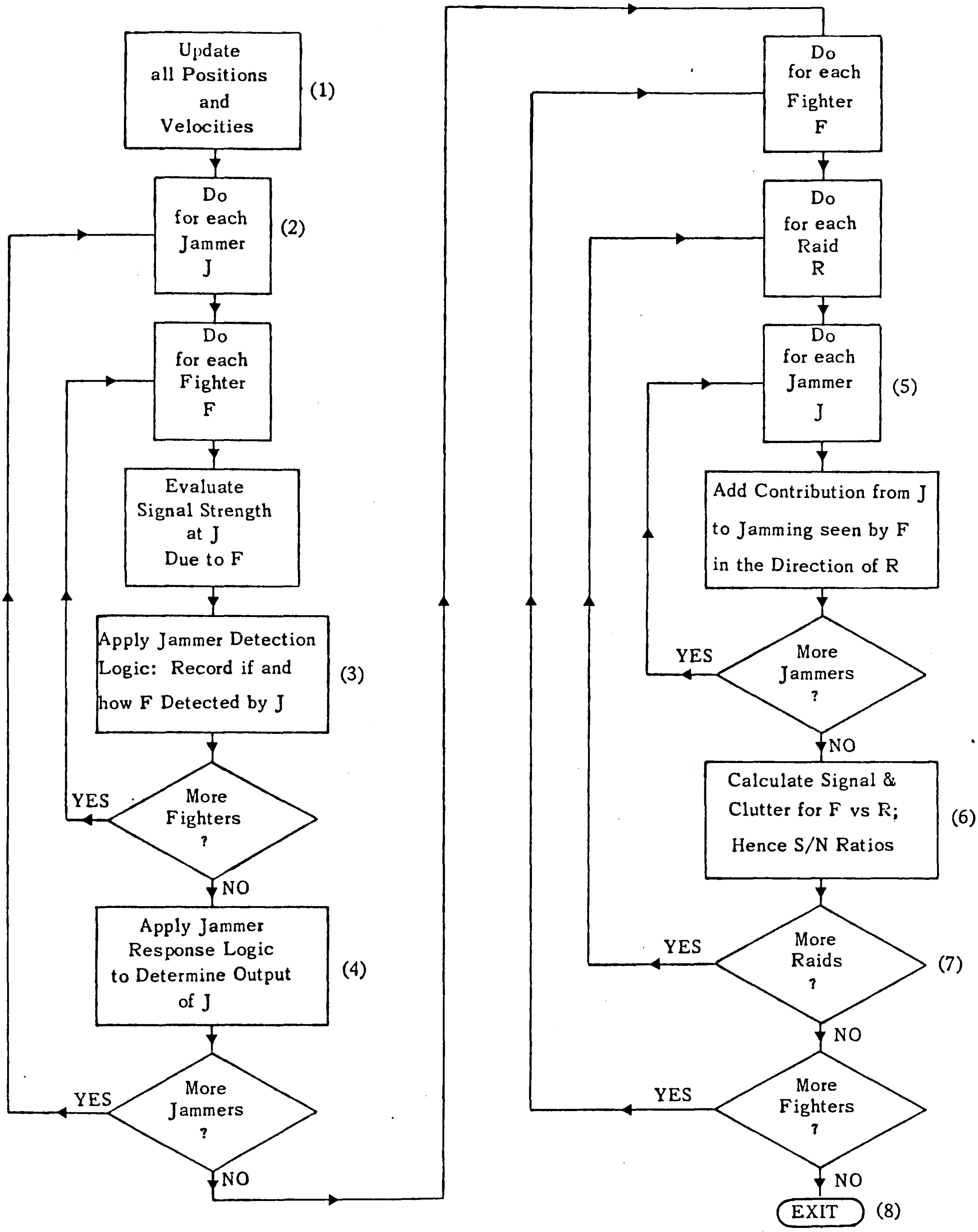


FIGURE 0.3. FLOWCHART FOR THE RADAR AND JAMMING ROUTINE

Jammer Logic

47. The jammer logic for responsive noise jammers is illustrated in figure 0.4; it shows how each jammer responds to each radar. (Stand-off jammers are assumed to emit continuous barrage jamming). A detection threshold must be specified in the input data for each jammer type. If the sidelobe strength is greater than this threshold, the radar is continuously detected and therefore continuously jammed. The sidelobe strength is specified in the input data simply as a power (in dB) down on the mainbeam strength. If the mainbeam strength is beneath this threshold, the radar is undetected and, unless it picks up jamming directed at another fighter, it is unjammed by this jammer. If the mainbeam only is greater than this threshold, and the radar is in a locked-on mode (see figure 0.5), then the radar will be continuously detected and therefore continuously jammed. If this is not the case, the radar is still in a scanning mode. Now, the input data must specify, for each jammer type, whether it can respond to scanning radars during their paint only. If the jammer has this capability, it jams the radar during its paint only; otherwise, this scanning radar goes unjammed by this jammer.

48. The principle of the jammer response is that a jammer concentrates its power into the minimum possible continuous frequency band. If just one radar is detected within the frequency band of a jammer transmitter it is spot-jammed, over a fairly narrow bandwidth (an input quantity). If several radars are continuously detected within the frequency band of one transmitter, the model assumes that jamming power is transmitted continuously over a bandwidth just encompassing the highest and lowest detected frequencies. Finally if the jammer detects any scanning radars, and it can respond to these during their dwell-time, if necessary it increases the above bandwidth to jam these radars during their dwell-time only.

FIGHTER RESPONSE LOGIC

49. For a given fighter and raid, the routines just described calculate at regular intervals the radar signal strength, perceived jammer strength, clutter strength and radar noise power. The ensuing fighter response to the raid, in view of these values, can best be described in terms of the values of a control variable MODER(IF), where IF denotes the fighter serial. This variable can take integer values from 0 to 13. The control variable MODEC(IF) is also useful; this is essentially a simplification of MODER(IF), taking values only from 1 to 4. The fighter response logic is based on the relationship between the various values of MODER(IF), which are as follows:

MODER(IF)=0 if fighter IF is allocated to scramble but has not yet taken off.

MODER(IF)=1 if fighter IF is on CAP or cruising to CAP.

MODER(IF)=2 if fighter IF has full information on its target raid, in clear conditions; it has locked-on to acquire range and track (ie not via a Track-While-Scan radar) and adopted an interception course.

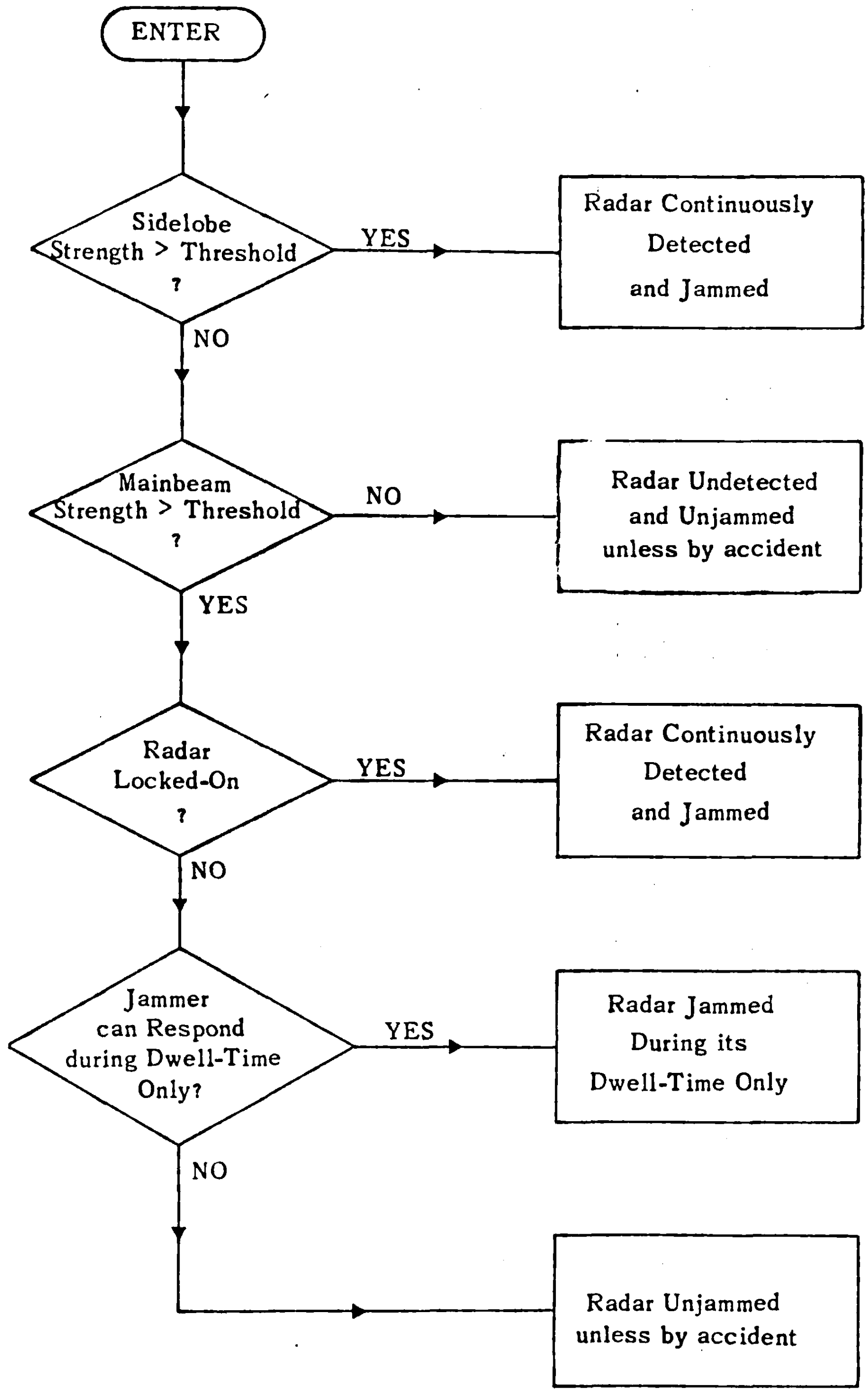


FIGURE 0.4. JAMMER LOGIC

MODER(IF)=3 if fighter IF has a radar missile in flight, in clear ECM conditions.

MODER(IF)=4 if fighter IF has reverted in the face of jamming to Home-on-Jam (HOJ) mode, from MODER(IF)=2 (ie full information with radar locked-on in clear conditions).

MODER(IF)=5 if fighter IF is on a Broadcast Control interception and has not yet detected a suitable target raid.

MODER(IF)=6 if fighter IF, in the face of jamming, has adopted a lead-pursuit course on its target.

MODER(IF)=7 if fighter IF has a radar missile in flight, in Angle-on-Jam (AOJ) mode.

MODER(IF)=8 if fighter IF has full information on its target raid in clear conditions via its Track-While-Scan (TWS) radar (ie it has not locked-on); it has adopted an interception course.

MODER(IF)=9 if fighter IF has reverted in the face of jamming to HOJ mode, from MODER(IF)=8 (ie full information via TWS radar in clear conditions).

MODER(IF)=10 if fighter IF has a radar missile in flight, in Home-on-Jam (HOJ) mode.

MODER(IF)=11 if fighter IF is on a Close Control (CC) interception and has not yet detected its target.

MODER(IF)=12 if fighter IF is on a Data Link (DL) controlled interception, and has not yet detected its target.

MODER(IF)=13 if fighter IF, originally on a CC or DL control interception, has detected its target but not yet acquired range information, and is continuing on the CC or DL interception course rather than adopt an autonomous lead pursuit course.

The values which the control variable MODEC(IF) may take are as follows:

MODEC(IF)=1 if fighter IF is on CAP or assigned to fly to CAP.

MODEC(IF)=2 if fighter IF is on, or assigned to, a BC interception.

MODEC(IF)=3 if fighter IF is on, or assigned to, a CC or DL interception.

MODEC(IF)=4 if fighter IF is, after achieving a detection, attacking a raid under autonomous control.

50. Figure 0.5 shows the sequence of events for a fighter from the moment when AI detection occurs to the point at which a missile is launched. Equivalently, it shows the relationships between the various values of MODER(IF) and MODEC(IF). The combination of multiple ECCM facilities and human behaviour in the cockpit makes this period complex and unpredictable

in detail, and the representation in the fighter model must inevitably simplify the situation considerably.

51. On its first entry to the response logic routine, a fighter always has MODER(IF)=1,5,11 or 12; similarly MODEC(IF)=1,2 or 3, depending on the figher's allocated role. After achieving a suitable detection MODEC(IF) is set equal to 4 to represent the assumption of autonomous control by the fighter. From this point onwards the fighter response, represented by the value of MODER(IF), depends upon whether it has a TWS radar, whether it is currently in clear conditions or in HOJ or AOJ mode, and whether it is within missile launch-range. Finally, if a fighter on a CC or DL interception course detects its target but does not burnthrough, it may, depending on input data, continue on its original course or adopt a lead-pursuit course.

52. The following notes enlarge upon some of the features of figure 0.5:

(1) For the purposes of this flowchart, close control and data link control need not be distinguished.

(2) When a fighter changes to autonomous control, Ground Control is informed to enable any intercept controller to be freed for other tasks.

(3) The model treats Track-While-Scan (TWS) radars differently from normal radars, for the former need not lock-on to acquire range. In the model the fighter is deemed to have acquired range on its target when the signal-to-total noise ratio exceeds an input threshold value; the threshold for TWS radar may differ from that for non-TWS radar.

(4) When a fighter does acquire range on its AI radar, it adopts a new interception course.

(5) It is conceivable that at some point the fighter's signal-to-total noise ratio is less than the TWS threshold but greater than the threshold required to lock-on and acquire range information. Nevertheless it is assumed that a fighter with TWS radar continues on its course until it can acquire range via TWS rather than attempt to lock-on to acquire range, which will normally provoke responsive jamming.

(6) This assumes that if a fighter has already been given an interception course by Ground Control (or via a Data Link), it will continue on this course until it acquires range, rather than adopt a lead-pursuit course. This tactical option is specified by the input data.

(7) The positions of all fighters and raids and their bearings from each other are periodically updated by the radar routine and at each entry to the radar routine this must be checked. This change of mode is carried out at missile launch, for radar missiles only. Radar missiles need to be guided during their flight so the monitoring of the prevailing ECM conditions, which affect the missile lethality,

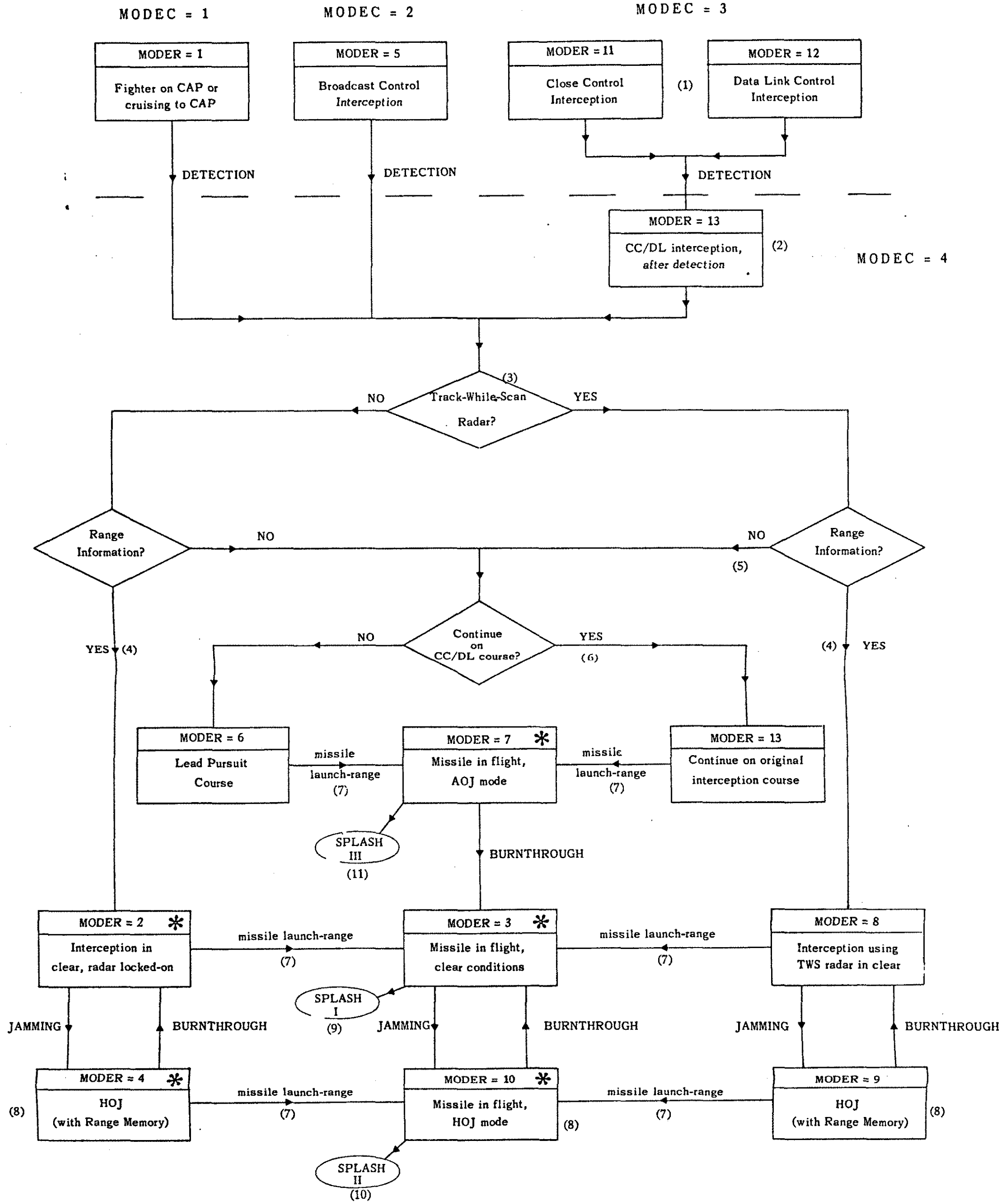
is continued. This is simplified in the model so that missile lethality can, for a given type of attack (forward or rear hemisphere), take one of three values only. These correspond to launch or splash in clear conditions, in HOJ mode (relying on memory of range information), and in AOJ mode (in which range information has never been available to the fighter's weapon system). It is assumed that the ECM conditions at launch affect the lethality of infra-red missiles.

(8) The signal strengths of all radars and jammers are periodically assessed by the radar routine, and jamming can therefore intervene at any time and divert the AI radar into the Home-on-Jam (HOJ) ECCM mode; conversely, burnthrough could be achieved. The criteria for these transitions are based on thresholds of the signal: jamming and jamming: noise strengths.

(9) Missile launch I represents a missile fired in clear (or near clear) ECM conditions, under which it should have maximum lethality.

(10) Home-on-Jam involves tracking the jamming target in angle only, using previously obtained values of range and closing velocity to estimate its current track. Although a fairly effective attack course can be flown (in the model the fighter continues to fly its original interception course), the method is subject to errors, which in the model are subsumed into the final missile lethality.

(11) This corresponds to launch in Angle-on-Jam mode, when the fighter has no range information at all. It gives the lowest missile lethality, because of the possibility that it is not actually within the launch envelope when its weapon system signals that launch-point is reached. (The determination of this point is based upon the rate of change of the bearing of the target from the fighter, and is subject to error).



Key
* These modes all imply locked-on radar, for jamming calculation purposes.

FIGURE 0.5. (i) FIGHTER EVENT SEQUENCE FROM AI DETECTION TO MISSILE LAUNCH
(ii) VALUE OF THE CONTROL PARAMETER MODER (IF)
(see text for notes)

EVENT PROCESSING OF FIGHTERS

53. There are two aspects of the event processing of fighters which merit special attention.

1. Directed Lists of Fighter Serials

54. The list of fighter serials is directed in order to facilitate searches for fighters IF, in the order in which they were first scrambled, within a specified range (MODE1,MODE2) of control mode MODEC(IF), ie fighters IF such that:

$$1 \leq MODE1 \leq MODEC(IF) \leq MODE2 \leq 4 \qquad (0.1)$$

The structure adopted also facilitates the elimination of a fighter serial from the list of active fighters if it runs out of fuel or ammunition. Its serial then becomes available for reallocation.

55. Each fighter serial IF has a successor IUP(IF) and a predecessor IDOWN(IF) such that:

$$IUP(IDOWN(IF))=IDOWN(IUP(IF))=IF \qquad (0.2)$$

The variable IFM(I) stores the serial IF of the first allocated fighter with MODEC(IF)=I, for I=1,2,3,4. If there is no fighter with control mode I, then IFM(I)=0. Finally, IFM(5) stores the first fighter serial available for allocation.

56. When a fighter is allocated for take-off it is henceforth identified by a serial; its prospective control mode K, which determines its proposed role, must be specified (K=1,2,3 or 4). The serial JF of the first fighter with control mode greater than K is then determined; if there are no such fighters then:

$$JF=IFM(5) \qquad (0.3)$$

The serial IF to be allocated is simply:

$$IF=IFM(5) \qquad (0.4)$$

The next available serial is updated by setting:

$$IFM(5)=IUP(IFM(5)) \qquad (0.5)$$

The required value of MODEC(IF) is then defined:

$$MODEC(IF)=K \qquad (0.6)$$

It is necessary to change the IUP and IDOWN pointers of IF and JF, so that IF is positioned at the end of the list of fighters in mode K, and points to the first fighter with control mode greater than K (or the first available fighter serial, if there are no such fighters). Finally, if

IF is the only fighter in mode K (so that IFM(K)=0) then IFM(K) is reset:

$$IFM(K)=IF \quad (0.7)$$

57. Conversely, when fighter IF with MODEC(IF)=K is removed from the list of active fighters the serial JF of the fighter immediately above IF in the list is obtained from:

$$JF=IUP(IF) \quad (0.8)$$

IF is removed from its current position in the active fighter list and placed below IFM(5). The serial IF becomes available for reallocation - in fact it becomes the first such serial - by setting:

$$IFM(5)=IF \quad (0.9)$$

Finally, if IF was the first fighter in mode K, IFM(K) must be reset. If IF was the only fighter in mode K then it is set equal to zero, otherwise it is set equal to JF.

58. Consideration of fighters with control mode K within a specified range (MODE1,MODE2), so that:

$$1 \leq MODE1 \leq K \leq MODE2 \leq 4 \quad (0.10)$$

is similarly straightforward. The serial IF1 of the first such fighter is found (if IFM(K)=0 for $MODE1 \leq K \leq MODE2$, then IF1=0). If $IF1 \neq 0$ subsequent fighters within the required range of control modes are obtained simply from the IUP array. The last such fighter is IDOWN(JF), where JF is the first fighter with control mode greater than MODE2 (or IFM(5), the first available fighter serial, if there are no such fighters).

2. Cancellation of Fighter Events

59. Every event in the fighter model is uniquely identified by three parameters - its type, the time at which it is due to occur, and the entity (raid or fighter) to which the event refers. Every event (except planned take-off) which refers to a fighter, namely event types 8,9,10,11, 12,14,16,17 or 18, can be cancelled if required. For example, if a fighter detects a raid it may abandon its cruise to or on CAP (event type 9 or 10) and instead adopt an interception course or a lead-pursuit course; when a raid changes track all fighters attacking it cancel their previous interception courses and adopt new ones, etc.

60. When an event for fighter IF is first generated, a pointer to the event data is stored in one of the variables LOCEVF1(IF), LOCEVF2(IF) or LOCEVF3(IF). This enables the event data to be easily accessed if it should later be necessary to cancel the event. When such an event is cancelled, or when it actually occurs, the relevant pointer is reset to zero. A simple test at the beginning of the event-cancelling routine then prevents the possibility of a time-wasting search through the event list to delete an event which no longer exists.

61. Three variables suffice for this purpose because at most only three events can be simultaneously pending for a fighter. The set of fighter events may be partitioned into the following three mutually exclusive sets, with the given one-one correspondence between sets and pointers:

(i) LOCEVF1(IF) points to either event type 7 (take-off) or event type 8 (change of profile-point), exactly one of which must be pending for fighter IF.

(ii) LOCEVF2(IF) may point to an event type 9 (arrival on CAP), type 10 (arrival at a subsequent CAP point), type 11 (arrival back at base), type 12 (arrival at expection interception point), or type 14 (recalculate fighter interception course after a raid track-change), at most one of which may be pending. If none of these events are pending then the fighter is on a lead-pursuit course and LOCEVF2(IF)=0.

(iii) LOCEVF3(IF) may point to an event type 16 (missile-splash), type 17 (test for missile-launch range), or type 18 (end of fighter turn after an attack), at most one of which may be pending; otherwise LOCEVF3(IF)=0.

MODELLING ASSUMPTIONS

62. Useful applications of the model to air defence problems will depend on the acceptability in any particular case of the assumptions incorporated. For convenient reference these are summarised below in two groups; first those which are fundamental to the structure of the model and then those which are relatively easily changed to suit the representation required.

Structural Features

(1) Enemy raids do not respond actively to being attacked, except in the case of jamming. They do not inflict damage on the fighters, and escort fighters are not represented. (This particular assumption is discussed in more depth in paras 67-70).

(2) Enemy raids (which can consist of a single aircraft) are considered as single points, except for those calculations described in para 14.

(3) The model is essentially two-dimensional, elevation angles not being represented.

(4) Information concerning enemy raids is generated in three stages: initial detection, knowledge of raid track parameters, and finally the number of aircraft in the raid.

(5) Of the radar types which occur in an air defence environment, only fighter AI radar is explicitly represented.

(6) The performance of AI radars is generally expressed simply in terms of S/N, J/N and S/J ratios.

(7) In general, once an initial radar detection has occurred, the corresponding target is assumed to be detected continuously thereafter, if only as a jamming spoke.

(8) Communications jamming, CW-beam jamming and 'deception ECM' are not represented.

(9) Fighters are allocated to raids on a 'least-time-to-intercept' basis.

(10) The internal components of a control centre are not modelled explicitly, though their effects, in terms of communications delays, accuracy of information and command decisions, are represented in simplified form.

(11) Except as regards defence resource limitations, the assignment of fighters to each enemy raid is basically independent of the assignment to every other raid - the whole air defence force is not 'optimised'.

(12) Air bases are treated simply as sources of fighters, and are characterised by position and fighter generation rate only.

(13) Close and broadcast control are only distinguished implicitly by their effects on fighter courses - the actual transmitted messages and information are not represented.

(14) The detailed flight-paths of air-to-air missiles are not represented - only the time of flight and the terminal lethality.

(15) It is assumed that identification (IFF) is perfect.

Non-Structural Features

(16) Enemy raid tracks consist of a series of straight sections at constant speed and altitude.

(17) Bombers and specialist escort jammers are assumed to be equally likely targets.

(18) Lock-on is assumed to occur when the signal-to-noise ratio exceeds a threshold value (specified in the input data).

(19) Initial allocations of fighters to targets are not changed unless interceptions become impossible.

(20) The particular jammer logic chosen for the model is described in paras 47-48.

(21) In-flight refuelling is not represented.

(22) The order in which enemy raids are detected is used to define their order of priority as targets.

(23) It has been implicitly assumed that several responsive jammers can operate near each other without mutual interference.

EXTENSION OF THE MODEL

63. The model as it stands can make a contribution to defence studies involving fighters. As for any further development of the model, the following are some of the more obvious extensions which promise to be cost-effective, in terms of the programming effort involved and the increased scope and applicability of the model.

Fighter Allocation

64. An important feature of the model is that the allocation to each enemy raid is carried out independently (subject to resource limitations, of course), ie the air defence force is not 'optimally matched' to the total threat. While this is probably a reasonable approximation to what actually would happen, it does take a rather narrow view of the total situation, and several simple modifications could be incorporated in future versions of the model:

a. The desired fighter: enemy aircraft ratio, specified in the input data, could be reduced as fighter resources become depleted, so as to maintain some sort of reserve capability.

b. The ratio could be modified to bias allocation in favour of raids near their targets.

c. Flexibility could be increased by not insisting that fighters necessarily adopt the least time to intercept course.

d. The density of fighters at any particular location could be used to influence their likelihood of being selected for an interception.

Measures of Effectiveness

65. In a model in which fighters suffer no losses, the obvious measure of effectiveness is the number of enemy aircraft shot down. The following factors may also be of interest in any particular study:

a. The remaining weapon load of returning fighters. If, for example, this was too high, it could indicate that fighters were running out of fuel before they had used up all their missiles. (Possible remedies could include carrying additional fuel tanks, or reducing the speed limit on interceptions). It could also suggest a poor missile mix, or a poor balance between front and rear hemisphere attacks.

b. The mean time to interception. An increased interception speed would reduce this time, but would increase fuel consumption and reduce the overall fighter endurance.

Part 2 - p.31.

Para. 66.b should be extended as follows.

The most immediate extension of the model should be to represent fighter and raid flight profiles in three dimensions rather than two, i.e. fighter and raid altitude should be modelled directly. This will allow more detailed fighter flight profiles to be specified in the input data, and it will enable the full capability of the clutter routines described later to be incorporated in the radar and ECM calculations. It will also allow for more detailed specification of missile performance, as a function of fighter and raid relative height, aspect angle and velocity.

c. The mean time spent searching for targets. This quantity is a function of AI radar performance, CAP positioning and the altitude and intensity of the enemy attack, and may represent a significant expenditure of fuel.

Some Technical Points

66. This paragraph briefly mentiones some extensions to the model of a more technical nature:

a. The representation of information received by a fighter via the reversionary modes of kinematic ranging and dog-leg ranging, in a heavy ECM environment, could be included.

b. The missile launch envelopes and corresponding time of flight envelopes are at present functions of target aspect angle only. If the data are available this could be extended so that they were also functions of raid and launch altitude, and aircraft velocities.

c. When in Home-on-Jam mode, the missile lethality could be allowed to be a function of enemy aircraft spacing. The lethality of some radar homing missiles can be seriously degraded if the jammers fly at a critical separation.

Fighter-Enemy Aircraft Interaction

67. There is considerable scope in the model to increase the responsiveness of the enemy to attack by fighters; the following paragraphs consider some factors which could be included.

Escort Fighters

68. A simple method of representing the effects of escort fighters is as follows. Up to the launch of the first AA missile by an attacking fighter, nothing is changed by the presence of escort fighters, though one of these may be shot down in mistake for a bomber. Subsequently, however, fighters of the two sides meet and after a suitable period of combat, not represented in detail, fighters of one side or the other emerge victorious, either to continue escorting the raid or to continue attacking it. A simple way of representing the combat is to pair off each fighter of the smaller force with one from the larger force, assessing the result of each 1:1 combat using a random number, taking into account the appropriate capabilities of the aircraft.

69. One could of course represent escort fighters in the same detail as air defence fighters, and allow dogfights, etc to develop accordingly. This would seriously increase the complexity of the model, probably increasing the run-time by a factor of two.

Evasive Manoeuvres by Enemy Bombers

70. If evasive action by the bombers is to be represented, the modelling penalties of doing so depend upon the degree of interaction between the bomber and attacking fighter. If the evasive response of a bomber is entirely automatic, the modelling task is considerably easier than if each aircraft has to be continuously modelled to decide what is to happen next. Any predetermined manoeuvres by the enemy can already be catered for by the model.

PART 2.2

DESCRIPTION OF THE MODEL SUBROUTINES

1. MASTER SEGMENT

The MASTER segment contains very little processing. Subroutine INPUT reads all the input data and performs all processing which is independent of the replication machinery - calculation of the times of each raid track change and the times at which each raid crosses each warning line, etc. Subroutine REPINIT carries out all the initialisation and event generation necessary at the beginning of each replication, while subroutine EVENTS conducts all event processing, including the subsequent generation of events, associated with that replication. After the final replication subroutine OUTPUT calculates and prints statistics on a number of random variables, data on which has been accumulated (in subroutine STATS) during the model run.

Data items are input in the units most commonly used for those items, as shown in the illustrative data file (Annex B). These are then converted to the internal program units - radians, metres, seconds, pounds, watts and MHz.

2. BLOCK DATA SEGMENT

This segment sets the values of constants used in the model - π, 4π, $c/4\pi$ and $\pi/180$ (denoted by DEGRAD), together with constants used only in the clutter calculations. These are NSEARCH, the number of points at which the range of a pulse doppler clutter integral is searched for singularities; E, the level of accuracy required in determining these singularities; RE, the effective earth's radius and NINTP, the number of intervals into which the integration range is subdivided in the numerical calculation of pulse and pulse doppler clutter integrals.

3. SUBROUTINE COLLISION

This routine is called from subroutine INTCALCONTROL to calculate the expected interception point of a fighter with a raid. When range and track information on a target become available in the model, either to a fighter or to Ground Control, a straight-line interception course, normally involving accelerated flight, is calculated for the fighter. Time delays for communication and for the fighter to scramble or achieve the correct heading are allowed for in subroutine INTCALCONTROL, as are errors in the estimate of the raid track. These factors affect the values of the following parameters, which are calculated in subroutine INTCALCONTROL:

(RELX,RELY): the estimated initial raid position relative to the fighter;

(RAIDVX,RAIDVY): the estimated initial components of the raid velocity;

RVT : the estimated initial raid speed.

These parameters are illustrated in Figure 3.1.

The interception geometry is illustrated in Figure 3.2. Using vector notation the raid is initially at $\underline{r}_0$ relative to the fighter, with velocity $\underline{v}$. Without loss of generality we may take the initial time t=0. The problem is to find a fighter course to give minimum-time interception. The interception profile for this fighter must already have been specified so that its distance-time relationship, denoted by $s(t)$, is known. Here flight consists of a series of constant acceleration segments. If the direction of this minimum-time interception is given by a unit vector $\underline{e}$, the interception occurs when t is such that

$$\underline{e}s = \underline{R_0} + \underline{v}t \tag{3.1}$$

ie when

$$s^2 = R_0^2 + 2\underline{R_0}.\underline{v}t + v^2t^2 \tag{3.2}$$

where

$$R_0 = |\underline{R_0}|$$
$$v = |\underline{v}|$$

Now, suppose the interception occurs during the i-th leg of the profile. Reset the time-origin to t_i, the time at the beginning of this leg; t_i is the time relative to the start of the interception at t_0 =0 and is known. Let T = time to interception after the beginning of this leg, so that the distance travelled by the fighter in interception is

$$s = s_i + V_i T + \tfrac{1}{2} f_i T^2 \qquad (0 \leq T < t_{i+1} - t_i) \tag{3.3}$$

where s_i, V_i are its distance travelled and speed at the beginning of this leg (both known) and

$$f_i = \frac{V_{i+1} - V_i}{t_{i+1} - t_i} \tag{3.4}$$

is its acceleration on the leg. Note that s_i, V_i and f_i are all non-negative.

With this change in notation, equations (3.1) and 3.2) show that interception occurs on the i-th leg when T is such that

$$\begin{aligned} \underline{e}s &= \underline{R_0} + \underline{v}(t_i + T) \\ &= (\underline{R_0} + \underline{v}t_i) + \underline{v}T \end{aligned} \tag{3.5}$$

ie when

$$s^2 = |\underline{R_0} + \underline{v}t_i|^2 + 2(\underline{R_0} + \underline{v}t_i).\underline{v}T + v^2T^2 \tag{3.6}$$

Note that $|\underline{R_0} + \underline{v}t_i|^2$ and v^2 are both positive, though the term in T, $2(\underline{R_0} + \underline{v}t_i).\underline{v}$, may be negative.

If we define

$$\phi = |\underline{R}_o + \underline{v}t_i|^2 + 2(\underline{R}_o + \underline{v}t_i).\underline{v}T + v^2T^2 - (s_i + v_iT + \tfrac{1}{2}f_iT^2)^2 \qquad (3.7)$$

then interception occurs when $\phi=0$. We therefore must solve a quartic in T for the least positive root. The coefficients of this quartic are as follows:

$T^4: -\frac{1}{4}f_i^2$	always negative or zero	
$T^3: -f_iv_i$	always negative or zero	
$T^2: v^2 - v_i^2 - s_if_i$	positive or negative	(3.8)
$T: 2(\underline{R}_o + \underline{v}t_i).\underline{v} - 2s_iv_i$	positive or negative (negative for an approaching target)	
$T^0: |\underline{R}_o + \underline{v}t_i|^2 - s_i^2$	always positive, otherwise interception would already have occurred.	

This quartic is set up, examined, and if appropriate solved, for each leg of the profile, until at least one positive root appears, ie until interception becomes possible. If there is more than one positive root on that leg, the smallest is chosen.

Bearing in mind the sign ambiguities, Descartes rule of signs gives the following four cases to be considered in the search for roots of the quartic:

Case	Sign of the coefficient T^4	T^3	T^2	T	T^0	Number of sign changes	
(i)	-	-	+	+	+	1	≤1 positive root
(ii)	-	-	+	-	+	3	≤3 positive roots
(iii)	-	-	-	+	+	1	≤1 positive root
(iv)	-	-	-	-	+	1	≤1 positive root

We also know that since $\phi_{T=0}$ is positive and the coefficient of T^4 is negative, there is at least one negative root and one positive root. Thus except for case (ii) above, there is exactly one positive root of the quartic, which is best located by Newton-Raphson iterative approximation from t_{i+1}, since $\phi.\phi''$ is positive there.

Case (ii) is more complicated; it corresponds to

$$v^2 - v_i^2 - s_i f_i > 0$$

and (3.9)

$$2(\underline{R}_o + \underline{v} t_i).\underline{v} - 2 s_i v_i < 0$$

If it is possible for there to be more than one root in the segment, approach from the left in the iterative approximation is essential, and $\varphi''|_{T=0}$ must be positive for guaranteed Newton-Raphson convergence.
In fact

$$\varphi''|_{T=0} = 2\left(v^2 - v_i^2 - s_i f_i\right) > 0 \qquad (3.10)$$

Figures 3.3 to 3.6 provide more specific illustrations of the conditions corresponding to each of the above four cases.

The representation of the above algorithm in the fighter model is illustrated in Figure 3.7 and the following notes refer to details of this flowchart.

(1) STØ corresponds to s_i, the distance travelled at the beginning of each profile-leg.

(2) R2Ø represents the factor $|\underline{R}_o + \underline{v} t_i|^2$, so that its initial value is $|\underline{R}_o + \underline{v} t_o|^2 = |\underline{R}_o|^2$ $(t_o = 0)$. V2 is the square of the raid speed, while RVØ represents the factor $(\underline{R}_o + \underline{v} t_i).\underline{v}$; its initial value is $\underline{R}_o.\underline{v}$ $(t_o = 0)$.

(3)(a) The fighter type is denoted by IFTY and it is assumed that profile 2 is used for the interception. The serial, INIT, of the last profile point reached is already known and the examination of the fighter profile to determine on which leg the interception occurs begins at this point. The variable TØ corresponds in the above notation to t_i .

(b) TSTARTS takes into account the fact that the fighter may begin its interception between profile points.

(c) On return to subroutine INTCALCONTROL, IMPOSS=0 or 1 if interception is possible or impossible respectively.

(4) In the above notation,

TNEXT=TPPT(2,IFTY,I+1) corresponds to t_{i+i}

V1	= VPPT(2,IFTY,I+1)	"	"	v_{i+i}
DT	= TNEXT-TØ	"	"	$t_{i+i}-t_i$
ACC	= FPPT(2,IFTY,I)	"	"	f_i

(3.11)

(5) At the beginning of the calculation, if the fighter has positive acceleration the origin (t_0, V_0) is taken to be the current fighter position (TØ,VØ) - see Figure 3.8:

$$(DT)_{new} = (DT)_{old} - \frac{(V1-V\emptyset)}{V1-VPPT(2,IFTY,INIT))} \tag{3.12}$$

$$T\emptyset = TSTARTS - TNEXT - (DT)_{new}$$

(6) The coefficient of T^2,T and T^o, denoted respectively by CO2,CO1 and CONST, are given by expressions (3.8)

(7) If the acceleration on this leg, ACC, equals zero, then from (3.8) the coefficients of T^4 and T^3 are zero so that the quartic reduces to a simple quadratic equation. If this possesses a valid solution the expected interception point can be derived immediately from the calculated time to interception; if not, the variable IMPOSS is set equal to 1 and control is returned to subroutine INTCALCONTROL.

(8) This test distinguishes case (ii), in which there may be 1 or 3 roots in a segment, from cases (i),(iii) and (iv), in which there is at most one root in this segment.

(9) To determine whether it is necessary to conduct a Newton-Raphson iteration in this segment the constant term CONSTNEXT of the quartic corresponding to the next segment is calculated; an iteration is carried out only if CONSTNEXT is negative. Certain preliminary variables must be calculated:

(a) $$R21 = |\underline{R}_0 + \underline{v}\, t_{i+1}|^2 \tag{3.13}$$

Now

$$R2\emptyset = |\underline{R}_0 + \underline{v}\, t_i|^2$$

so that

$$\begin{aligned} R21 - R2\emptyset &= |\underline{R}_0 + \underline{v}\, t_{i+1}|^2 - |\underline{R}_0 + \underline{v}\, t_i|^2 \\ &= (t_{i+1} - t_i)[2\, \underline{R}_0 . \underline{v} + |\underline{v}|^2 (t_{i+1} + t_i)] \\ &= (t_{i+1} - t_i)[2(\underline{R}_0 + \underline{v}\, t_i).\underline{v} + |\underline{v}|^2 (t_{i+1} - t_i)] \end{aligned}$$

ie

$$R21 - R2\emptyset = DT * [2.0 * RV\emptyset + V2 * DT] \tag{3.14}$$

(b) $$RV1 = (\underline{R}_0 + \underline{v}\,t_{i+1}).\underline{v} \tag{3.15}$$

Now

$$RV\emptyset = (\underline{R}_0 + \underline{v}\,t_i).\underline{v}$$

so that

$$RV1 - RV\emptyset = \underline{v}\,(t_{i+1} - t_i).\underline{v}$$

ie

$$RV1 - RV\emptyset = |\underline{v}|^2 (t_{i+1} - t_i) \tag{3.16}$$

(c) $$ST1 = s_{i+1} \tag{3.17}$$

Now

$$ST\emptyset = s_i$$

so that

$$ST1 - ST\emptyset = s_{i+1} - s_i$$

ie

$$ST1 - ST\emptyset = (t_{i+1} - t_i).\frac{(v_i + v_{i+1})}{2} \tag{3.18}$$

Then from expression (3.8):

$$\text{CONSTNEXT} = |\underline{R}_0 + \underline{v}\,t_{i+1}|^2 - s_{i+1}^2$$

ie

$$\text{CONSTNEXT} = R21 - (ST1)^2 \tag{3.19}$$

(10) If CONSTNEXT is positive, no root of the quartic occurs on this profile-leg. The variables t_i, v_i, $|\underline{R}_0 + \underline{v}t_i|^2$, $(\underline{R}_0 + \underline{v}t_i).\underline{v}$ and s_i are all updated and the next profile-leg is considered.

(11) In the unlikely event CONSTNEXT = 0 interception occurs just as the fighter reaches the end of this profile-leg. In this case the variable T is set equal to DT=$(t_{i+1} - t_i)$ and the time to interception TINT calculated from

TINT=TØ+T-TSTARTS

The coordinates (PX,PY) of the expected interception point relative to the fighter's initial position may then be found immediately.

(12) This carries out a Newton-Raphson iteration, with the initial approximation at the left-hand end of the profile-leg. If a root, T, is found the time to interception and the expected interception point may be calculated immediately.

(13) This carries out a Newton-Raphson iteration, with the initial approximation at the right-hand end of the profile leg.

FUNCTION DBTOABS

This Function is called from subroutines INPUT and GAINREAD to convert input data expressed in decibels to their corresponding absolute values.

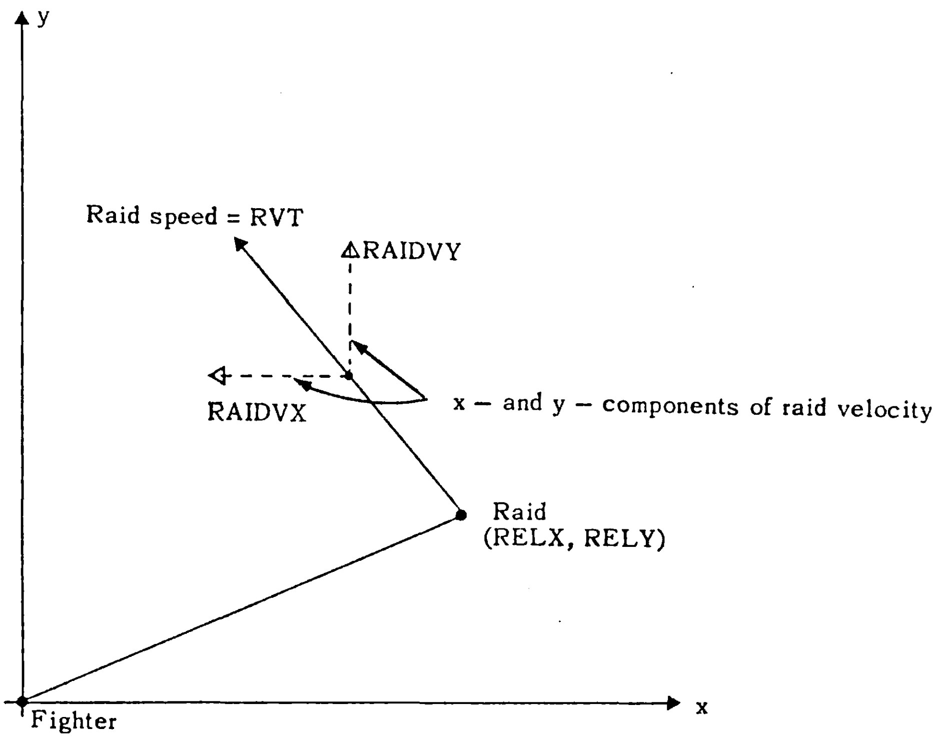


FIGURE 3.1. INITIAL RAID PARAMETERS – SUBROUTINE COLLISION

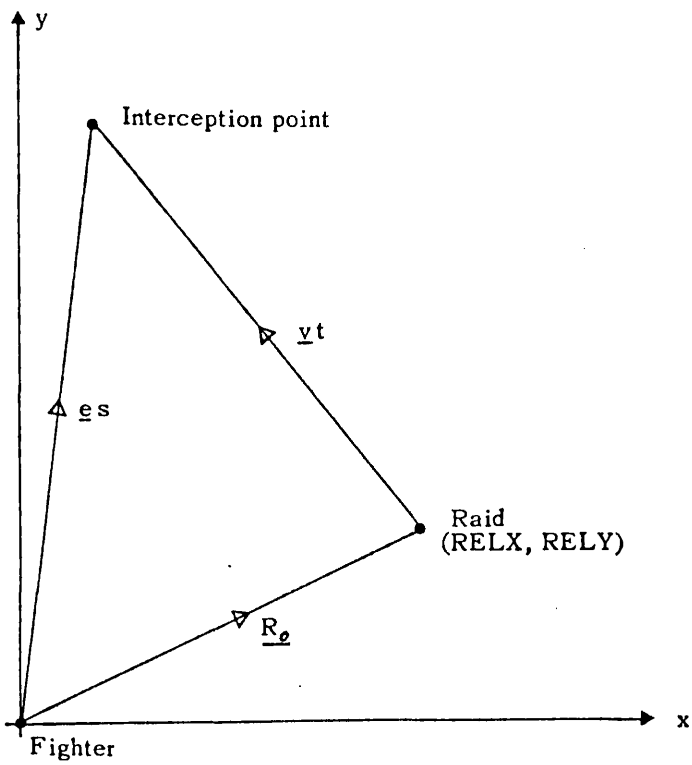
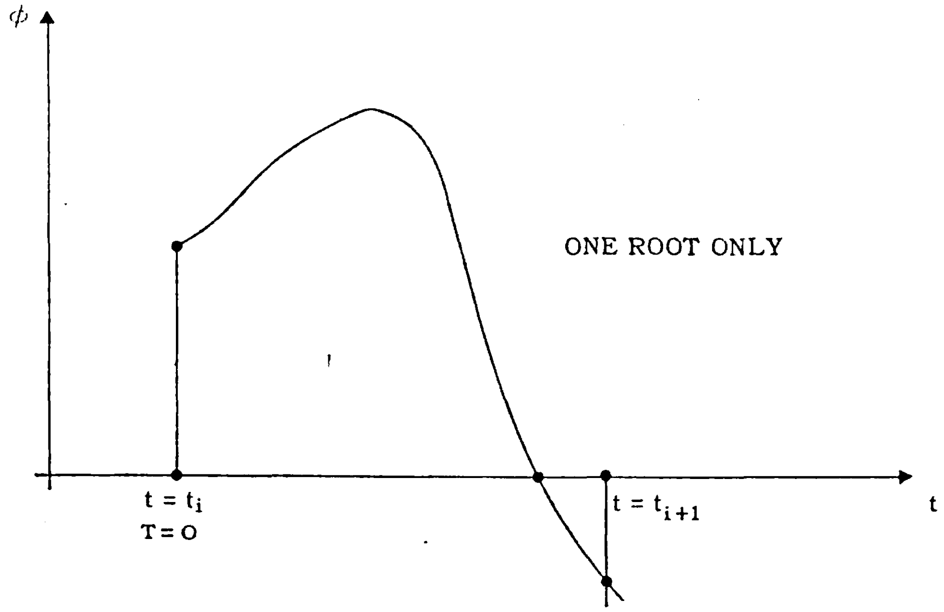
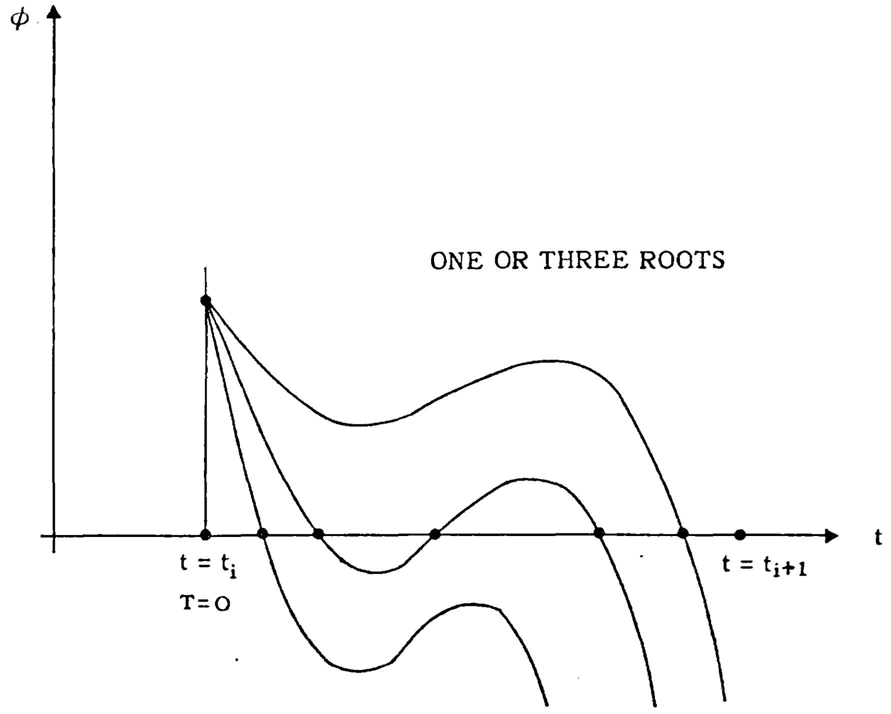


FIGURE 3.2. INTERCEPTION GEOMETRY – SUBROUTINE COLLISION



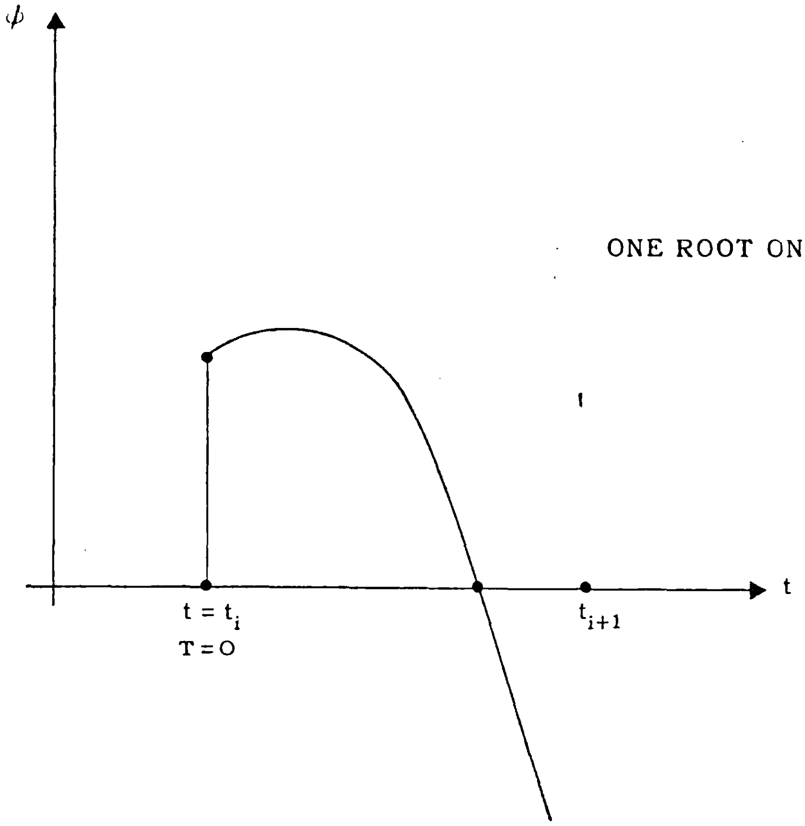
Iterate from t_{i+1} if $\phi(t_{i+1})$ is negative to find the root

FIGURE 3.3. CASE (i) ϕ, ϕ', ϕ'' AT T = O ALL POSITIVE – SUBROUTINE COLLISION.



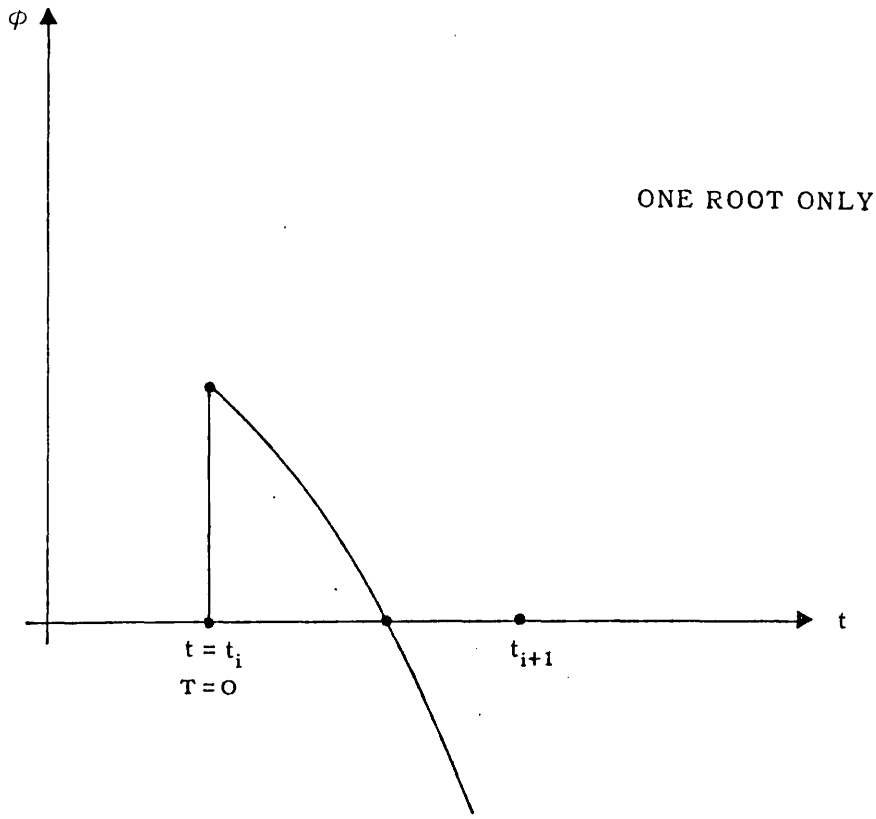
Iterate from t_i. If no root, check $\phi(t_{i+1})$; if it is negative, iterate from t_{i+1}.

FIGURE 3.4. CASE (ii) ϕ, ϕ'' POSITIVE, ϕ' NEGATIVE AT T = O – SUBROUTINE COLLISION



Iterate from t_{i+1} if $\varphi(t_{i+1})$ is negative to find the root

FIGURE 3.5. CASE (iii) ϕ, ϕ' POSITIVE, ϕ'' NEGATIVE AT T = O – SUBROUTINE COLLISION



Iterate from t_{i+1} if $\phi(t_{i+1})$ is negative to find the root

FIGURE 3.6. CASE (iv) ϕ POSITIVE, ϕ', ϕ'' NEGATIVE AT T = O – SUBROUTINE COLLISION

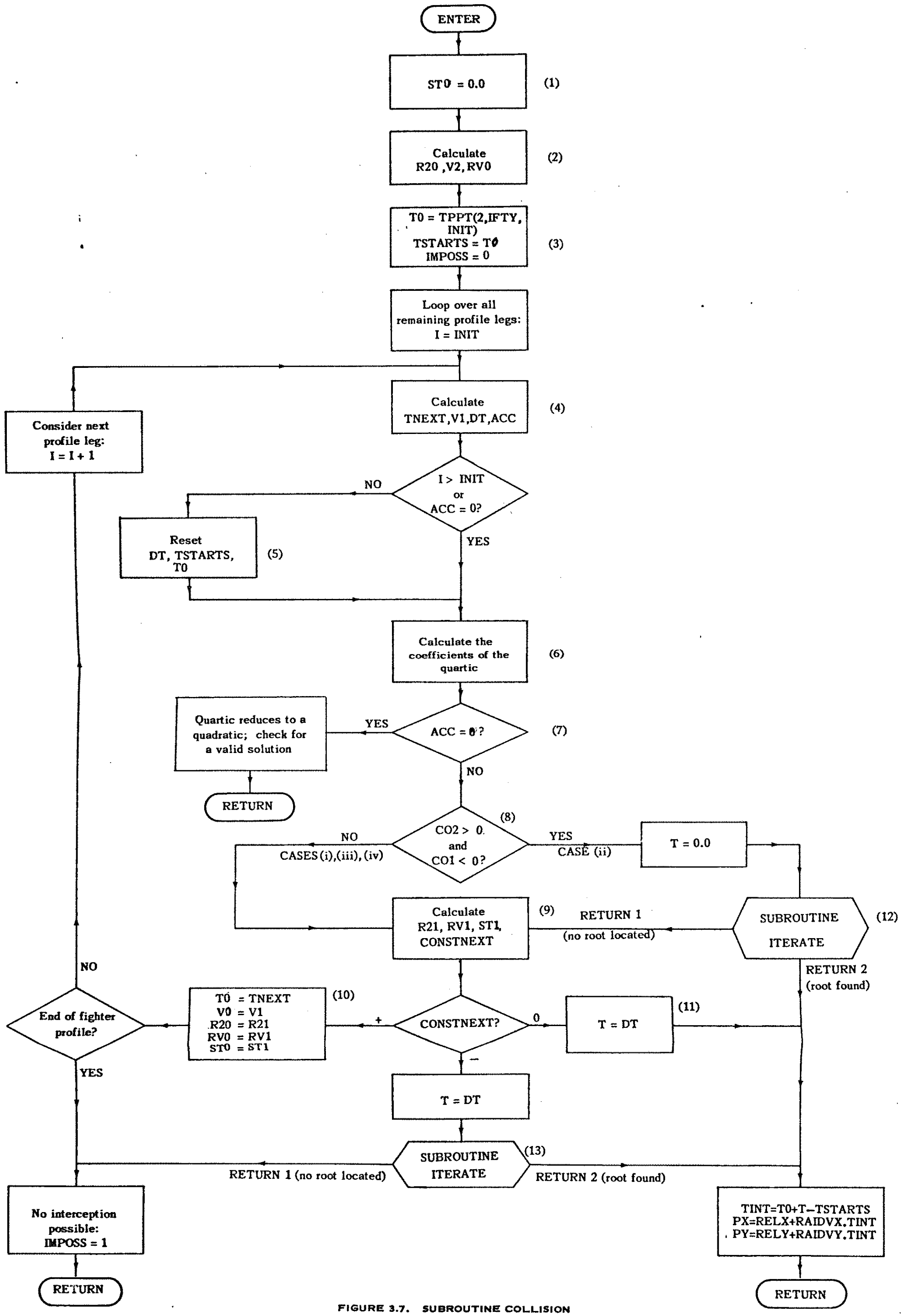


FIGURE 3.7. SUBROUTINE COLLISION

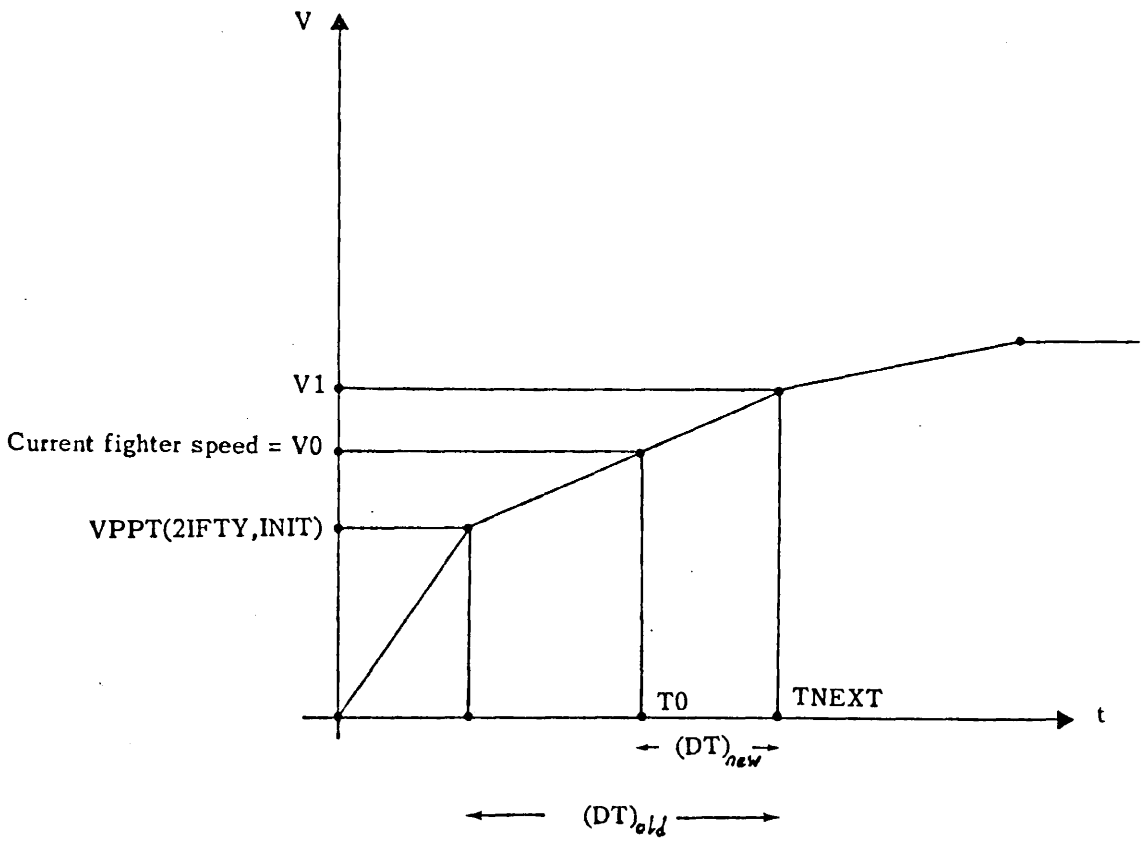


FIGURE 3.8. CALCULATION OF THE ORIGIN (T 0 , V 0) - SUBROUTINE COLLISION

4. SUBROUTINE DELETE(LOSE)

This routine is called to delete an event for fighter IF from the event queue. When any such event which may later be deleted is first inserted into the event queue. the pointer to the event data is stored in one of the variables LOCEVF1(IF), LOCEVF2(IF) or LOCEVF3(IF); otherwise these pointers are always set equal to zero. The pointer to the event which is to be deleted is brought into this routine in the parameter list, and is denoted here by LOSE. If LOSE=0, the event has either already occurred (in which case the pointer is reset to zero in subroutine EVRESET) or it has already been cancelled;regardless, no action is taken. Otherwise, subroutine DELEVT is called with parameter LOSE to delete this event from the event queue. Subroutine DELETE then resets the pointer LOSE to zero and control returns to the calling routine.

The above three variables suffice to refer to all events pending for fighter IF, since at most three can be pending simultaneously (see subroutine EVRESET). Thus these variables are not restricted to referring to unique events; they may each refer to one of a set of mutually exclusive events. The variable LOCEVF2(IF) may point to an interception event (type 12), an arrival at a CAP point (event types 9 and 10), arrival back at base (event type 11) or an event type 14 (recalculation of fighter interception course after a raid track-change), since, for a given fighter, at any time only one of these event types may be pending. Similarly, LOCEVF1(IF) may refer to a fighter take-off (event type 7) or a change of profile - point (event type 8). LOCEVF3(IF) may refer to a pending missile-splash (event type 16), an event type 17 (missile launch-range test) or an event type 18 (fighter turning to carry out a further attack within the reattack sequence).

Subroutine DELETE is called in the following circumstances:

(i) from subroutine EVENTS, at event type 2. When a raid changes track, every fighter IF currently attacking that raid has a new interception course calculated, taking into account the fighter and Ground Control reaction delay. This processing is carried out at an event type 14. The interception event (type 12) is cancelled, while LOCEVF2(IF) is reset to contain the pointer to the event type 14 for fighter IF.

(ii) from subroutine EV12, which sets up an interception (event type 12) for fighter IF. The event to which LOCEVF2(IF) previously pointed is cancelled (ie type 9,10,11,12 or 14) and the new interception event inserted into the event list; LOCEVF2(IF) is reset to contain the pointer to the new event.

(iii) from subroutine EVENTS, at event type 16. When a fighter achieves a missile splash against a raid, the corresponding interception event, type 12,which represents the expected collision of the fighter and raid becomes irrelevant and so is cancelled.

(iv) from subroutine EV9, when a fighter is no longer able to continue with a planned interception. The interception event is cancelled and the fighter is either sent to CAP (event type 9) or returns to base (event type 11). LOCEVF2(IF) is reset to contain the pointer to the event type 9 or 11 respectively. Any possible missile-splash or launch-range test or reattack is also cancelled by a call to subroutine DELETE with parameter LOCEVF3(IF). Finally, the pending event type 8 (change of profile point) is cancelled by a call to subroutine DELETE with parameter LOCEVF1(IF). A new event type 8, corresponding to the next profile point due in profile 1, the cruise profile, is then generated either here or in subroutine EV8.

(v) Before a new event type 8 (change of fighter profile point) is set up in subroutine EV8, which may be called from a number of routines, the previous event type 8 which was pending for that fighter is always cancelled by a call to subroutine DELETE with parameter LOCEVF1 (IF).

(vi) When a fighter is deleted from the list of active fighters all of its pending events are cancelled by calls to subroutine DELETE with parameters LOCEVF1(IF), LOCEVF2(IF) and LOCEVF3(IF).

(vii) If in subroutine RADEVENTS a fighter adopts a lead-pursuit course, it cancels any planned interception event (or possibly, if it was previously cruising to or on CAP, an event type 9 or 10).

(viii) Within the reattack sequence (subroutine RAIDKILL), if fighter IF's target is destroyed by another fighter its planned interception and possible missile launch or splash are cancelled. An attempt is then made to find another target for the fighter in subroutine REATTACK.

(ix) In subroutine REATTACK interception courses are calculated against all targets to determine which raid, or flight within a raid, can be attacked most quickly. If a fighter is within a reattack sequence without having achieved burnthrough, attacking in AOJ mode, the choice of next target is retained although the corresponding interception event is cancelled. The fighter then continues on a lead-pursuit course until burnthrough is achieved or it launches a missile in AOJ mode.

(x) Finally, it is also assumed in subroutine REATTACK that after an attack a fighter turns onto a new attack heading at constant angular velocity. The pending event type 8 (change of profile point) for this fighter is then delayed by the time taken to complete this turn.

5. SUBROUTINE DELEVT(LOSE)

This routine is called from subroutine DELETE to delete an event relating to fighter IF from the event queue. (None of the possible raid events - track-change or crossing a warning line - may be cancelled, unless the raid is totally annihilated, since raid evasive manoeuvring is not represented; see subroutine RAIDELIM).

When an event for fighter IF which may later be deleted or cancelled is first inserted into the event queue in subroutine PUTEVT, the pointer to the event data is returned to the calling routine and stored in one of the variables LOCEVF1(IF),LOCEVF2(IF) or LOCEVF3(IF). Then, if this event must later be deleted, the search for its location in the queue is greatly simplified. Also when such an event either occurs or is cancelled, the relevent pointer - LOCEVF1(IF),LOCEVF2(IF) or LOCEVF3(IF) - is reset to zero, in subroutine EVRESET or DELETE respectively. A test on the value of this pointer at first entry to subroutine DELETE therefore ensures that the event queue is never searched unnecessarily to cancel an event which has not been generated or has already been cancelled.

The pointer to the event which is to be cancelled is brought into subroutine DELETE in the parameter list. If this is already zero no action is taken, otherwise it is brought into subroutine DELEVT, as the parameter LOSE. The relevant event is then deleted from the queue; the cell thus emptied is returned to the free-cell stack and the next-free-cell pointer is updated. Finally, on return to subroutine DELETE the relevant pointer is reset to zero.

6. SUBROUTINE DEVIATION

This routine is called from subroutine OUTPUT to calculate the mean, standard deviation of the mean, standard deviation and variance of a random variable X over n replications, where $\sum_{i=1}^{n} X_i$ and $\sum_{i=1}^{n} X_i^2$ are known (X_i is the value of X on replication i).

7. SUBROUTINE DOTPROD(X,Y,VX,VY,ANGLE)

This subroutine may be called from subroutines COLLISION, JPOWER,LAUNCHTEST, RADAR and SOJPOWER. It determines the angle θ ($0 \leq \theta \leq \pi$),as illustrated in Figure 7.1. The vectors $\underline{r}$ (X,Y) and $\underline{V}$ (VX,VY) are specified in the parameter list. (Any translation of axes, so that the point P is effectively at the origin of coordinates as shown in the Figure, is carried out in the calling routine). The equation used is

$$\underline{r}.\underline{V} = (X.VX+Y.VY) = |\underline{r}||\underline{V}| \cos\theta \qquad (7.1)$$

The angle θ is denoted by ANGLE in this subroutine and is returned in the parameter list.

8. SUBROUTINE ERRORS

If the model is running in stochastic mode (MODERUN=1) this routine selects random variables PE,QE,RE and SE from a N (0,1) distribution to be applied to errors in the estimates of the raid parameters (x-coordinate, y-coordinate, speed and heading respectively) in the calculation of fighter intercept courses. If the model is running in deterministic mode (MODERUN=0), thesefour variables are all set equal to 1.0 at the beginning

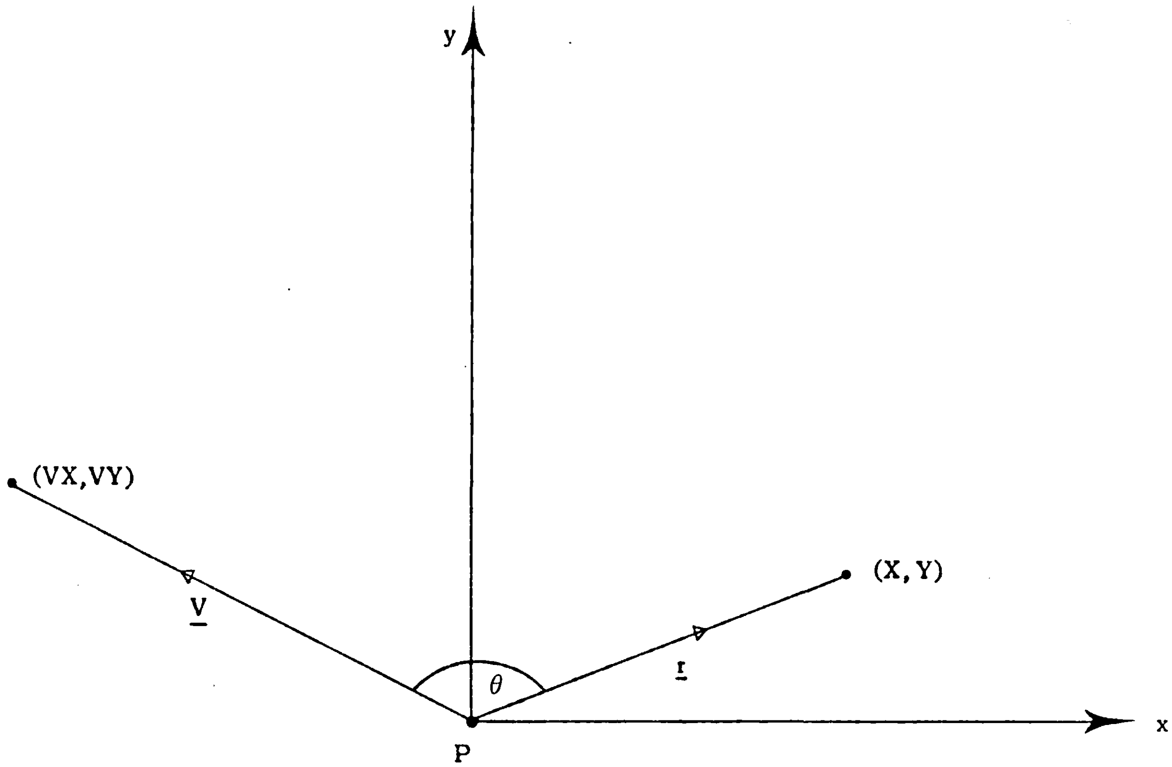


FIGURE 7.1. DERIVATION OF THE ANGLE θ - SUBROUTINE DOTPROD

of the run in subroutine INPUT and no action is taken in this routine.

9. SUBROUTINE EVENTS

This is the primary routine in the Fighter Model and is called from the MASTER segment once per replication. All the processing associated with every event generated is either executed in this routine or in subroutines called from this routine.

Preliminary Processing

The parameters T, IEV and IF which define the next event to be processed are retrieved from the event list via subroutine GETEVT. T is the time at which the event is due to occur, IEV is the event type and IF (or IR) is the serial of the fighter (or raid) to which the event refers. (Two events, types 1 and 6, do not refer to specific fighters or raids, so that IF=IR=0). Subroutine EVPRINT prints these three event parameters, while subroutine EVRESET resets, if necessary, one of the three fighter control parameters LOCEVF1(IF),LOCEVF2(IF)or LOCEVF3(IF). (These are used to simplify the event cancellation procedure; they store pointers to pending fighter events, so that they may be easily located in the event list and then deleted. In particular, if a fighter event is not deleted and actually occurs, the relevant pointer must be reset to zero). The clock is updated (ie the variable TIME is set equal to T) and the trace switched off or on as necessary (controlled by the input parameters TRACESTART and TRACEEND). The processing for the event proper can then begin. There are currently 18 different event types; the processing associated with each event is reached via a COMPUTED GO TO statement.

Event Type 1 - End of Replication

This event is generated at the beginning of each replication in subroutine REPINIT, and it terminates the replication. Although most scenarios should come to a natural conclusion it is useful to be able to stop a replication at a pre-determined time.

Event Type 2 - Raid Track-Change

This event either executes or generates all the processing associated with a raid track-change. For convenience the starting-point of a raid is defined as a track-change, so that the first event type 2 for each raid is generated at the beginning of each replication in subroutine REPINIT.

The raid serial IR having already been determined, subroutine NEWTRACK is first entered. This routine resets the kinematic raid parameters (position, heading, speed and height) and sets up the next track-change. (If the raid has reached the end of its final track-leg it is 'deactivated' by setting LOCEVR(IR)=0). On returning from subroutine NEWTRACK the variable INOCHANGE is checked to determine if the raid track has actually changed, for the weapon-release point must be specified as a track-change point and the raid need not necessarily alter speed, height or course after weapon-release. If there is no track-change or the raid has reached the end of its pre-defined track, no action is taken and the program returns to examine the next event. Similarly, if the event type 4 for this raid has not been generated, so that no track information is available to Ground Control (GC) the track-change effectively remains undetected so that again no action is taken.

If there are fighters already attacking this raid (NATTACK(IR)>0) subroutine ERRORS is called. In stochastic mode this selects random variables PE,QE,RE and SE from a N(0,1) distribution; in deterministic mode they remain at their initial value of 1.0. These variables are used later to derive the errors in the estimates of raid parameters when fighter interception courses are recalculated. This processing is carried out at event type 14; to ensure that the same error estimates apply to all fighters IF which are currently attacking this raid, these random variables are stored:

$$\begin{array}{l} PERR(IF)=PE \\ QERR(IF)=QE \\ RERR(IF)=RE \\ SERR(IF)=SE \end{array} \qquad (9.1)$$

(The serial IF of each fighter attacking the raid is recovered from the IATTACK array).

The event type 14 (interception course recalculation) for each fighter IF attacking the raid must be set up. If the fighter already has a missile in flight against the raid, this processing is ignored. (The variable LOCEVF3(IF), if non-zero, is a pointer to the next event for fighter IF of type 16,17 or 18, for these cannot be pending simultaneously. If it is of type 16, ie a missile-splash, then IEVENT(2,LOCEVF3(IF))=16).

Subroutine RESPACC determines the total delay TDELAY suffered by fighter IF before it can adopt a new interception course; this comprises a GC processing delay plus a fighter response delay. The expected interception of fighter IF (event type 12) is then cancelled by a call to subroutine DELETE with the control variable LOCEVF2(IF) as the parameter. The event type 14 for fighter IF is then generated, to occur a time TDELAY hence. This processing is repeated for each fighter IF attacking the raid.

Finally, if the event type 4 for this raid has occured, so that IEV4(IR)=1, then GC has track information on the raid and an event type 20 is generated to occur a time TRESPONSER(IR) hence, where TRESPONSER(IR) is the GC delay in processing track-change information on raid IR. This event determines if there is any shortfall in the number of fighters assigned to attack the raid and, if necessary, it selects and allocates extra fighters to intercept the raid. If the appropriate event type 4 has not occured (IEV4(IR)=0)GC does not yet have track information on the raid and the event type 20 is not generated.

Event Type 3 - Raid first detected by Ground Control

This event is generated for each raid at the beginning of each replication in subroutine REPINIT and corresponds to the crossing of warning line 1, ie the initial detection of the raid. (The times at which the three warning lines are crossed - if they are crossed - by each raid are calculated in subroutine INPUT, since they are constant for all replications, and the corresponding events (types 3, 4 and 5) are set up in subroutine REPINIT).

It is assumed that when the raid is first detected a specified number NFØ (NFØ≥0) of fighters is scrambled to fly to a CAP; the CAP allocated depends on the serial IR of the raid, and is specified in the raid input data. It is also assumed that the fighters at every base start decreasing their readiness levels towards the minimum readiness level when the first event type 3 occurs, ie when the first raid is detected. A variable TEV3 (set equal to -1 in subroutine REPINIT) stores the time of the first event type 3. Under this assumption subroutine FCHOOSE selects and scrambles, from any base, the NFØ fighters which can most quickly reach the CAP. After take-off these fighters search for enemy raids on their AI radars.

Event Type 4: Ground Control acquires raid track information

This event represents the Ground Control (GC) response to the acquisition of track information on raid IR. Track information on a raid can be generated in one of two ways, the first of which is more usual:

(i) The raid crosses warning line 2. The time at which this occurs is calculated in subroutine INPUT and the corresponding event type 4 generated at the beginning of each replication in subroutine REPINIT.

(ii) A fighter scrambled when a raid is first detected (event type 3) may acquire track information on a raid before it crosses warning line 2. In this case an equivalent event, type 13, is generated.

Note that the raid track and warning lines may be defined in the input data so that an event type 3 occurs but event types 4 and 5 do not, to represent, for example, spoof raids.

The model represents errors in the estimates of raid position, speed and heading in the calculation of interception courses, and a delay in the GC response to a raid track-change. Subroutine UPDATEERROR is called to assign values to these estimates and this GC reaction delay. If the raid has already been detected and tracked by a fighter, the source - GC or fighter - providing the smallest such errors or shortest delay is assumed to be chosen. (If a conflict arises between the sources of information providing the smallest errors and the shortest reaction delay, a criterion must be specified in the model to determine which of these factors takes precedence).

Apart from the single call to subroutine UPDATERROR(IR,0) if the raid crosses warning line 2, the same processing is carried out if the track-change information is generated by GC (event type 4) or by a fighter (event type 13). In particular, the variable IEV4(IR), initialised in subroutine REPINIT, is used to ensure that all the following processing for raid IR is carried out once only.

If the raid has been completely destroyed or has reached the end of its defined flight path, then LOCEVR(IR)=0 and no more processing is necessary. Otherwise, subroutine MOVER is called to update the raid's position. Each fighter already attacking the raid - if any - has its position and velocity updated in subroutine MOVE1F and a new interception course calculated in subroutine COLLISION (via subroutine INTCALCONTROL). If a fighter possesses a valid interception course this is assumed to be instantly adopted; the fighter's change of track is set up in subroutine EV12 (via subroutine INTCALCONTROL). If a fighter cannot intercept the raid it is removed in subroutine IATTDEL from the list (IATTACK(IR,·)) of fighters attacking raid IR and provisionally sent to CAP in subroutine EV9. Subroutine ERRORS is first called before any of these fighter interception courses are calculated; this ensures that the same errors in the estimates of raid position, speed and heading apply to all the fighters.

The input data specifies a number, NF1, of fighters which should attack a raid once its track has been ascertained. If, after new interception courses have been calculated for all fighters currently attacking the raid there is any shortfall, subroutine FFIND finds those fighters cruising to or on CAP which can intercept the raid most quickly. Finally, if sufficient such fighters cannot be found, subroutine FCHOOSE attempts to meet the remaining shortfall by scrambling unallocated fighters from the fighter bases.

Event Type 5 - Raid Size Resolved

Event type 5 represents the Ground Control response to the acquisition of accurate information on the strength of a raid, ie the number of aircraft within it. As with event type 4, an event type 5 can be generated in one of two ways:

(i) The raid crosses warning line 3. The time at which this occurs is calculated just once in subroutine INPUT, and the event type 5 for each raid is then set up at the beginning of each replication in subroutine REPINIT.

(ii) A fighter scrambled at an earlier event type 3 or 4 could possibly resolve the raid on its AI radar before it crosses warning line 3.

It is assumed that the event type 4 (GC acquisition of track information) for a raid must occur before the corresponding event type 5. The input data could of course be such that the size of a raid is never established. As with event type 4 a variable, IEV5(IR), is used to ensure that the following processing is only carried out at most once for each raid. Again, it is possible that raid IR is completely destroyed before it reaches warning line 3 (in which case LOCEVR(IR) is set equal to zero in subroutine RAIDKILL).

The input data contains a parameter, FRATIO, which represents the ratio of fighters: enemy aircraft which GC attempts to achieve when the strength of a raid is known. After updating the raid's position the number of fighters required to attack this raid is calculated. If there is a shortfall subroutine FFIND attempts to meet this with the fighters which can intercept the raid most quickly from those cruising to or on CAP. If insufficient such fighters are available subroutine FCHOOSE attempts to meet the remaining shortfall from the unallocated fighters on alert at the fighter bases.

Event Type 6 - AI Radar Scan

This is the most complex event in the Fighter Model. Its frequency of occurrence is determined by a parameter DTR in the input data. The first event type 6 is generated at the beginning of each replication in subroutine REPINIT; each event type 6 then generates its successor, unless all raids have been completely destroyed.

Event type 6 corresponds to the updating, every DTR seconds, of the complete radar and ECM picture for each airborne fighter and each raid. If there are no fighters remaining active - ie with sufficient fuel and missiles on-board to be capable of an interception - or none have yet taken off, no processing is carried out other than setting up the next radar scan. Otherwise the positions and velocities of all fighters and raids are updated in subroutines MOVEF and MOVER respectively. Subroutine RADAR then updates the radar and ECM picture for each fighter and raid pair. In particular it determines, for each fighter and raid, the raid signal strength at the fighter, the total jamming experienced by the fighter when it is illuminating the raid and the fighter's background noise and clutter levels. Subroutine RADAR then calls subroutine RADEVENTS to determine what should be the fighter's response to this raid in the face of these values of target signal strength, jamming strength, noise and clutter levels.

Event Type 7 - Fighter take-off

This event, for fighter IF, is generated in subroutine FCHOOSE when a fighter is allocated either to fly to a CAP (at an event type 3) or to intercept a raid (at an event type 4 or 5, or to replace a fighter whose planned interception is for some reason no longer possible). The current profile of fighter IF is recovered from the KPROF array, the fighter type from the KFTYPE array and the base from which this fighter is operating from the KBASE array. (A profile is a set of values of velocity and fuel-consumption rates at a series of time-points. The fighter profile is initially set in subroutine FCHOOSE, and is determined by its expected task. It is conceivable that this task could be changed in the interval between the fighter allocation and take-off; for example a raid could change track, rendering a planned interception impossible; conversely a raid could cross warning line 2 or 3 so that more fighters are allocated to attack it. At such occurrences the fighter profile is changed accordingly).

The variable NFIGHTCUM is updated; this is simply a count of the number of fighters which take-off. (It is initialised at the beginning of each replication in subroutine REPINIT). The fighter's fixed AI carrier frequency is then set. If the model is running in stochastic mode its value is chosen from a uniform distribution within the minimum and maximum carrier frequencies (specified in the input data). When in deterministic mode the carrier frequency, F, of fighter n scrambled (n=1,2,3,......) is given by

$$F(n) = \begin{cases} A+\frac{n}{2}B & (n=2,4,6,......) \\ A-\frac{(n-1)}{2}B & (n=1,3,5,.......) \end{cases} \qquad (9.2)$$

Here A is the average carrier frequency for this fighter type and B is the maximum systematic frequency difference for this fighter type, calculated in subroutine INPUT by

$$A=\frac{\text{Max. frequency}+\text{Min. frequency}}{2} \tag{9.3}$$

$$B=\frac{\text{Max. frequency}-\text{Min. frequency}}{21}$$

(The denominator 21 is the current maximum number of fighters expected to be airborne at any one time, and equals the dimension of all fighter arrays. This can easily be changed of course).

The value of the control variable MODER(IF) is next determined. The control variable MODEC(IF) will already have been set equal to 1,2 or 3 depending on whether the fighter is to fly to a CAP, attack a raid under Broadcast Control (BC), or attack a raid under either Close Control (CC) or Data Link Control (DL) respectively. This is too 'coarse' a variable for many purposes, taking only four possible values (MODEC(IF)=4 corresponds to fighter IF attacking a raid autonomously, after achieving a detection). MODER(IF) currently takes one of 14 values, with a correspondence with MODEC(IF) as illustrated in Figure 0.5. (MODER(IF)=0 corresponds to a fighter between allocation and take-off, regardless of the value of MODEC(IF)).

The variable FT(IF) is updated, the queue for take-off at the fighter base decreased by one and the fighter's next profile-point change (event type 8) set up in subroutine PUTEVT.

Event Type 8 - Fighter reaches a profile point

The fighter position, velocity and fuel state are updated in subroutine MOVE1F, while the serial of the last profile point reached is retrieved from the variable JPPT(IF). The time until the next profile point is due is determined from the TPPT array and the next profile-point is set up in subroutine PUTEVT. The fighter acceleration components are then reset; FFX(IF) and FFY(IF) denote the current x- and y- components of fighter IF's acceleration. FPPT(JPROF,IFTY,M) stores the magnitude of the acceleration between profile points M and M+1 for fighter type IFTY under profile JPROF. If fighter IF has reached a constant speed FFX(IF) and FFY(IF) are unchanged - they are already zero - and no action need be taken. Otherwise, its new components are altered simply in proportion to the magnitude of the change in acceleration.

Finally the current fighter fuel consumption rate, FRATE(IF), and the serial of the last profile point reached,JPPT(IF), are updated.

Event Type 9 - Fighter arrives on CAP and
Event Type 10 - Fighter reaches a CAP point

Each CAP in the Fighter Model is defined by a series of points, the coordinates of which are specified in the input data. These points define the sequence of straight line segments which constitute the CAP. The processing carried out at event types 9 and 10 is identical. A distinction is drawn only in order to make the arrival of a fighter on CAP distinct from its ensuing progression through the CAP points.

An event type 9 is generated either in subroutine FCHOOSE for a fighter scrambled to fly to a CAP, or in subroutine EV9 if a fighter's planned interception is rendered impossible and it is reallocated to a CAP. The fighter position and fuel state are first updated in subroutine MOVE1F. The fighter type, the serial of this CAP and the serial of the CAP point at which the fighter is arriving are retrieved from the KFTYPE,KCAP and NEXTCAPPT arrays respectively. NPOINTS(JCAP) denotes the number of points defining this CAP. The next CAP point is updated and the distance DS to this point found from the arrays XPT and YPT; the fighter velocity components are then reset to correspond to the fighter flying along the straight line segment towards the next CAP point.

It is expected that by the time a fighter reaches a CAP it will have finished accelerating and reached a constant speed and constant fuel consumption rate section of its flight profile. Nevertheless, to ensure that DT, the time taken to fly to the next CAP point, is always correctly evaluated, subroutine TFROMS is called; this takes into account any remaining acceleration between profile points. Finally subroutine PUTEVT sets up an event type 10, the next arrival at a CAP point, at time DT hence.

Event Type 11 - Fighter returns to Base

This event is generated in subroutine EV9 if fighter IF, unable to continue an interception, cannot reach any of the pre-defined CAP patterns. While flying back to base the fighter may detect and attack enemy raids on its AI radar. If no suitable targets present themselves to the fighter, on arrival back at base it is deleted from the list of active fighters (represented by a call to subroutine FDESTR1(IF)).

Event Type 12 - Fighter reaches its expected interception point

This event corresponds to fighter IF reaching its expected interception point and is generated in subroutine EV12.

The fighter and raid positions are updated by subroutines MOVE1F and MOVER respectively and the serial IR of the fighter's target raid recovered from the ITARGET array. Subroutine WRITERAD prints information on the fighter and raid positions and velocities and on the distance of the raid from its weapon-release point.

The remaining processing associated with this event depends on the value of the control variable I12OR16, specified in the input data.

(i) I12OR16=0

In this case the fighter's measure of effectiveness is event type 16-missile splash. The occurrence of this event type 12 means that fighter IF has reached its expected interception point before splashing a missile against raid IR. If the fighter has not even detected the raid (MODEC(IF)<4) the interception is abandoned and the fighter provisionally sent to CAP. An attempt is made to replace fighter IF in the attack on raid IR, in subroutines FFIND and (if necessary) FCHOOSE. Otherwise, if the fighter has detected the raid and is in the process of attacking it, this event type 12 is ignored: the fighter continues along its course, while event type 17's continue to be generated (in subroutine LAUNCHTEST) to test for missile-launch range.

(ii) I12OR16=1

If required, the fighter measure of effectiveness may be taken to be event type 12, ie interception or collision. In this case, when the fighter has acquired range information on its target on its AI radar and set up its own interception course, no further radar or ECM processing of this fighter is carried out. (This is achieved by setting ICOLLIDE(IF)=1 in subroutine RADEVENTS when the fighter achieves burnthrough; see subroutine TESTFIGHTER). Similarly, no tests for missile launch-range are made, and all missile data is ignored. In this case subroutine STATS collects interception data; for completeness ICOLLIDE(IF) is reset to its initial value of -1 and, currently, subroutines IATTFIND,IATTDEL and FDESTR1 eliminate fighter IF from the list of fighters attacking raid IR and the list of active fighters. Finally, it is possible that, with I12OR16=1 specified in the input data, fighter IF has been assigned an interception course by Ground Control and has reached its expected interception point without even detecting the raid. In this case subroutine STATS does not collect interception data and there is no need to reset ICOLLIDE(IF), since it has not been altered from its initial value, set at the beginning of each replication in subroutine REPINIT. Subroutines IATTFIND,IATTDEL and FDESTR1 still delete the fighter from the list of fighters attacking raid IR and the list of active fighters.

Event Type 13 - Ground Control acquires track information from a fighter

This event is generated in subroutine RADEVENTS when a fighter first achieves range and track information on raid IR. It is identical with event type 4 - track information on raid IR generated by GC when it crosses warning line 2 - except that subroutine UPDATEERROR(IR,0) is not called. (This subroutine call updates, as a result of the raid crossing warning line 2, the standard deviation of the errors in the estimates of raid position, speed and heading, and the reaction delay to any track-change by this raid).

Event Type 14 - Recalculate fighter interception course

This routine recalculates the interception course of fighter IF on raid IR, as a result of a track-change by raid IR. It is generated at the raid track-change event(type 2) to take into account the delay in the fighter's response to this information.

If the fighter has a missile in flight, no action is taken. Similarly if the fighter has adopted a lead-pursuit course (MODER(IF)=6) and either GC has no track information (IEV4(IR)=0) or the fighter prefers its own LP course to a GC interception course (IGVSF=0 in the input data), no action is taken. Otherwise the raid and fighter positions and velocities are updated in subroutines MOVER and MOVE1F respectively and the raid serial IR recovered from the ITARGET array.

The random errors PE,QE,RE and SE to be applied to the estimates of raid position, speed and heading in the fighter course recalculation in subroutine INTCALCONTROL are retrieved:

$$\begin{aligned} &PE=PERR(IF)\\ &QE=QERR(IF)\\ &RE=RERR(IF)\\ &SE=SERR(IF) \end{aligned} \tag{9.4}$$

The values PERR(IF),....,SERR(IF) are calculated at event type 2, the raid track-change event. This procedure ensures that the same errors in the estimates of the raid parameters apply to <u>every</u> fighter IF attacking raid IR when it changes track. Subroutine INTCALCONTROL calculates (via subroutine COLLISION) the fighter's new interception course; if the fighter has not yet taken-off the delay before take-off is taken into account in this calculation. If the fighter can still intercept the raid its new course is set up in subroutine EV12 from subroutine INTCALCONTROL and this concludes the processing. Otherwise, if as a result of the track-change the fighter can no longer intercept, it is removed from the list of fighters attacking raid IR by subroutines IATTFIND and IATTDEL and provisionally sent to a CAP in subroutine EV9. An attempt is made to replace fighter IF in the attack on raid IR, either from the fighters cruising to or on CAP (in subroutine FFIND) or, if necessary, from the unallocated fighters on alert at the fighter bases (in subroutine FCHOOSE).

<u>Event Type 15 - Spare</u>

<u>Event Type 16 - Missile Splash</u>

This event corresponds to the splash of a missile fired by fighter IF and is set up in subroutine LAUNCHTEST. The fighter and raid positions are updated by subroutines MOVE1F and MOVER respectively and the serial IR of fighter IF's target raid recovered from the ITARGET array. Any possible pending interception event (type 12) is cancelled by a call to subroutine DELETE with parameter LOCEVF2(IF); this variable is a pointer to the appropriate event information for fighter IF in the event list.

The missile probability of kill, P_k, is now calculated. This is a function of missile type (MTYPEF(IF), set up in subroutine LAUNCHTEST), ECM conditions (clear, Home-on-Jam or Angle-on-Jam) and the hemisphere of the attack (set up in subroutine LAUNCHTEST; IHEM(IF)=1 or 2 if the attack is forward or rear hemisphere respectively). If the fighter has fired a radar missile (MISSMODE(MTYPEF(IF))=1) the ECM conditions at splash are assumed to apply, since the missile needs to be guided during its flight; the control variable MODER(IF)=3,10 or 7 for clear conditions, HOJ or AOJ respectively. If it is an infra-red missile (MISSMODE(MTYPEF(IF))=2) the ECM conditions at launch are assumed to govern the missile P_k; this is represented by the variable IRLETH(IF), set up in subroutine LAUNCHTEST. Thus IRLETH(IF)=0,1 or 2 if the ECM conditions when fighter IF launches its missile are AOJ,HOJ or clear, respectively. The missile P_k, denoted by MPK1, may then be determined from the input array of kill probabilities, MPK.

Subroutine WRITERAD prints information on the fighter and raid positions and velocities. A random variable, Y, uniformly distributed between 0 and 1, is then generated. The missile P_k,MPK1, is compared with Y to determine, if the model is in stochastic mode, if a kill has been achieved. (In deterministic mode fractional numbers of enemy aircraft are allowed and exactly MPK1 aircraft will be eliminated from the raid). Subroutine STATS and RAIDKILL collect statistical data on the missile-splash and carry out the elimination of aircraft from the raid, unless no kill has been achieved in which case control jumps immediately to subroutine REATTACK. This subroutine chooses the next target, if any, for the fighter and sets up all the appropriate flight parameters. On return from subroutine RAIDKILL control again normally passes to subroutine REATTACK, unless this missile splash has finally annihilated raid IR (LOCEVR(IR)=0). In this case all fighters attacking the raid, including fighter IF, are provisionally sent to CAP in subroutine RAIDKILL. The further allocation (if any) of another target for this fighter is then carried out in subsequent calls to subroutine RADAR, rather than subroutine REATTACK, so that no further processing is required here.

Event Type 17 - Missile Launch check

This event tests at discrete intervals the range from fighter IF to its target, raid IR, to determine if it is within missile-launch range. If not the next event type 17 for this fighter is generated, while if it is within range a missile is launched and missile-splash event (type 16) is generated.

Subroutines MOVE1F and MOVER update the fighter and raid positions and velocities respectively, while subroutine LAUNCHTEST actually carries out all the processing associated with this event. Event types 17 and 16 are only generated if the input variable I12OR16=0, ie the fighter measure of effectiveness equals event type 16 or missile-splash, rather than simply event type 12 or interception/collision. In this case the first event type 17 is generated in subroutine RADEVENTS when the fighter first detects the raid; each event type 17 then generates its successor at intervals of DTLOOK (specified in the input data) in subroutine LAUNCHTEST.

Event Type 18 - Fighter released after a turn

It is currently assumed that the position of a fighter when turning between attacks on a raid remains constant. This event releases fighter IF at the end of such a turn, so that it may continue on its next straight-line interception course. The turn-time is calculated in subroutine COLLISION as part of the calculation of the time to the next interception.

The fighter's position is frozen in subroutine REATTACK by setting ITURN(IF)=1. (This also prevents any radar and ECM processing of the fighter during its turn - see subroutine TESTFIGHTER). Event type 18 is then generated in subroutine REATTACK, to correspond to the end of the turn.

The processing involved with this event is very straightforward. The variable ITURN(IF) is reset to zero, while an event type 17 is generated so that the fighter can begin testing the range to its target for possible missile-launch.

Event Type 20 - Raid Track-Change (subsidiary)

This event is generated by a track-change of raid IR (event type 2). It calculates the number of fighters currently required to attack the raid and attempts to meet any shortfall, from fighters cruising to or on CAP if possible (in subroutine FFIND) or, if necessary, from unallocated fighters on alert at the fighter bases (in subroutine FCHOOSE).

10. SUBROUTINE EVPRINT

This routine is called from subroutine EVENTS to print information on the next event pending. This information may be suppressed by appropriate choices of input parameters: TPRINT is the time before which no event information is printed; NOEVPRINT(L)>0 if this information on event type L is not to be printed at any time.

11. SUBROUTINE EVRESET(IEV,IF)

This routine is called from subroutine EVENTS immediately after each event is retrieved from the event list and before the processing of the event. IEV is the event type and IF the serial of the entity to which the event refers. If this event refers to a fighter (determined by the type of the event,IEV), this routine initialises one of the control variables LOCEVF1(IF),LOCEVF2(IF) or LOCEVF3(IF).

A test is made on the event type serial, IEV. If IEV=2,3,4,5,13 or 20 the event refers to raid IF and no action is taken. Similarly if IEV=1 or 6 this is the end of replication or radar scan event respectively and again no action is taken.

When an event for fighter IF is first generated a pointer to the event data in the event list is stored in one of the variables LOCEVF1(IF), LOCEVF2(IF) or LOCEVF3(IF). This enables the event data to be easily accessed if it should later be necessary to cancel the event. When such an event is cancelled or, as now, when it actually occurs, the relevant pointer is reset to zero.

Three variables suffice for this purpose because at most only three events can be simultaneously pending for a fighter. The set of fighter events is partitioned into the following three mutually exclusive sets, with the given one-one correspondence between sets and variables:

(i) If IEV=7 (fighter take-off) or 8 (profile-point change), exactly one of which must be pending, reset LOCEVF1(IF)=0.

(ii) If IEV=9 (arrival on CAP) or 10 (arrival at a subsequent CAP point) or 11 (arrival back at base) or 12 (arrival at expected interception point) or 14 (recalculate fighter interception course after a raid track-change), at most one of which may be pending (the fighter may be on a lead-pursuit course), reset LOCEVF2(IF)=0.

(iii) If IEV=16 (missile-splash) or 17 (test for missile-launch range) or 18 (end of fighter turn after an attack), at most one of which may be pending, reset LOCEVF3(IF)=0.

12. SUBROUTINE EV8(K,I)

This subroutine is called from subroutines FFIND and RADEVENTS (with K=2) and EV9 (with K=1) to set up the next event type 8 (change of profile-point) in profile K for fighter IF;I is the nearest profile-point in profile K with corresponding speed less than or equal to the current fighter speed. (I is evaluated in the calling subroutine). The time DT until the next profile-point is due is calculated by linear interpolation. Note that profiles are assumed to be monotonically increasing, so that if a fighter reaches a constant speed on a profile it continues at that speed as long as it remains in the profile.

LOCEVF1(IF) contains a pointer to the next event type 8 and this is cancelled by a call to subroutine DELETE with parameter LOCEVF1(IF). Subroutine PUTEVT then sets up the new event type 8 a time DT hence and stores the pointer to this event in LOCEVF1(IF).

13. SUBROUTINE EV9

This routine is called to provisionally send fighter IF to CAP if for some reason it can no longer continue with an interception. It may be called from the following subroutines:

(i) from RADEVENTS, if after acquiring range and track information a fighter finds it cannot intercept its target;

(ii) from LAUNCHTEST, if the fighter is too close to fire or has run out of missiles;

(iii) from REATTACK, if after a first attack a fighter cannot make any subsequent attacks;

(iv) from EVENTS at an event type 4, if a fighter attacking a raid under a lead-pursuit course is informed that it cannot intercept when Ground Control acquires track information;

(v) from EVENTS at an event type 14, if after a raid track-change a fighter can no longer intercept its target;

(vi) from EVENTS at an event type 12, if the fighter's interception course is so much in error that it has reached the expected interception point without detecting its target raid.

(viii) from RAIDELIM, if a raid has either been completely destroyed or has reached the end of its defined flight-path. All fighters attacking the raid are released for reallocation.

In all these cases, IF is deleted from the IATTACK array of the target raid (the list of fighters attacking it) in the calling routine.

The fighter control mode MODEC(IF) is set equal to 1, to correspond to a fighter cruising to or on CAP, by a call to subroutine FCHMODE. Similarly, if the fighter has already taken-off (MODER(IF)>0), the control variable MODER(IF) is also set equal to 1. The interception event and any possible planned missile-launch event are cancelled by calls to subroutine DELETE with parameters LOCEVF2(IF) and LOCEVF3(IF) respectively. (These are the pointers to the relevant event data in the event list). The nearest CAP and nearest point of entry to that CAP are found for the fighters and their serials, ICAP and IPT, are stored in KCAP(IF) and NEXTCAPPT(IF) respectively. The fighter profile, KPROF(IF), is set to 1, the cruise profile.

The time to CAP, DT, is calculated, directly from the TF array if the fighter has not yet taken off, by a call to subroutine TFROMS if it has not reached the maximum cruise speed, or by assuming an instantaneous drop in speed to the maximum cruise speed if it has. The fuel used in cruising to CAP, FLT, is also calculated. The serial JPPT(IF) of the last profile-point reached is altered to take into account the change from profile 2 to profile 1, and the next change of profile-point (an event type 8) in profile 1 is set up in subroutine EV8. The expected next profile-point in profile 2 is cancelled by a call to subroutine DELETE with parameter LOCEVF1(IF). The fighter change of course is represented by an appropriate change in the velocity and acceleration vectors, ie (FVX(IF),FVY(IF))and(FFX(IF),FFY(IF)) respectively. The fighter's fuel consumption rate, FRATE(IF), is altered to take into account the change of profile.

Finally, if the fighter has sufficient fuel to reach the CAP, the arrival on CAP a time DT hence (an event type 9) is set up in subroutine PUTEVT, which also stores a pointer to this event in LOCEVF2(IF). Otherwise a test is made that the fighter has sufficient fuel to return to a fighter base. If it can reach a base the variable KBASE(IF) is updated, the fighter immediately turns and flies towards the base and the corresponding event type 11 (fighter arrives back at base) is set up. On the flight to CAP or back to base the fighter's radar continues scanning and it is also available for reallocation, so it is possible that the fighter may detect or be allocated another raid to attack. If the fighter has insufficient fuel to fly to a CAP or to a base, subroutine FDESTR1 is called to cancel any pending events for the fighter and eliminate it from the list of active fighters.

14. SUBROUTINE EV12

This routine is called from subroutines INTCALCONTROL, FCHOOSE and FFIND to set up an interception event (event type 12) for fighter IF. Common Block INTERCEPT brings into this routine the coordinates (PX,PY) of the expected interception point relative to the fighter's current position; the expected time to interception, TINT, and the serial, INIT, of the last profile-point reached (in profile 2, the interception profile).

Subroutine RESPACC finds the fighter's heading error HEADERR and subroutine ROTATE then alters the values of PX and PY to take this heading error into account. The fighter change of track is represented simply by an appropriate change in the velocity and acceleration components, (FVX(IF),FVY(IF)) and (FFX(IF),FFY(IF)) respectively. Any pending event of the type whose pointer is stored in LOCEVF2(IF) (ie type 9,10,11,12 or 14 - see subroutine EVRESET) is cancelled by subroutine DELETE with parameter LOCEVF2(IF). Finally, subroutine PUTEVT sets up the interception event a time TINT hence, and stores the pointer to this new event in LOCEVF2(IF).

15. SUBROUTINE FCHMODE(IF,MNEW)

This subroutine changes the control mode, MODEC(IF), of fighter IF to MNEW (MNEW=1,2,3 and 4). It simply calls subroutine FDESTR(IF) to eliminate fighter IF, in its current mode, and then calls subroutine FCREA(IF,MNEW) to recreate it with MODEC(IF)=MNEW. It is called from subroutines EV9,FFIND,RADAR and REATTACK.

16. SUBROUTINE FCHOOSE(ICAP,NFNEC)

This routine is called to select a specified number, NFNEC, of fighters from the unallocated fighters waiting at various readiness levels at the various fighter bases. The integer ICAP specifies the proposed role of the fighters:if ICAP>0 FCHOOSE is called to scramble fighters to fly to a holding CAP, the serial of which is ICAP; if ICAP=0 FCHOOSE scrambles fighters in order to intercept a particular raid with serial IR. In both cases, the NFNEC fighters are chosen which can complete their mission - fly to CAP or intercept a raid - most quickly. The subroutine logic is illustrated in Figures 16.1 and 16.2; the following notes refer to details of this flowchart.

(0) If subroutine FCHOOSE has been called to scramble fighters to intercept a raid, so that ICAP=0, then subroutine ERRORS is first called. This ensures that in the subsequent calculation of interception courses (via subroutine INTCALCONTROL) the same errors in the estimates of raid position, speed and heading apply to all fighters.

(1) K is a count of the number of fighters required.

(2) TMIN1 will, eventually, equal the shortest time in which the specified mission can be accomplished (see(7)) so that it is initialised here as a very large number.

(3) Up to 3 fighter types may be specified. It is assumed here that they are input in the data file in a chosen order, for type 1 is selected in

preference to type 2, which in turn is always preferred to type 3. Similarly the (maximum) of 4 readiness levels are assumed input in increasing order, so that level i represents a shorter delay than level j ($1 \leq i < j \leq 4$). The best fighter type available at the base, IFTY, at the lowest readiness level, ILEV is chosen and these values, appertaining to base IBASE, are noted:

IFREP(IBASE)=IFTY

(16.1)

ILREP(IBASE)=ILEV

(4) The time before take-off, TONBASE(IBASE), is the maximum of the following three times:

(a) Initial fighter readiness level less the time since warning first received, ie

RLEV(ILEV)-(TIME-TEV3)

TEV3 is the time at which the first raid is detected, ie the time at which the first event type 3 occured.

(b) The minimum readiness level, RLEV(1).

(c) Time until the previous fighter scrambled has taken-off, ie

TBGO(IBASE)-TIME

TBGO(IBASE) is set to zero at the beginning of each replication in subroutine REPINIT and updated later in this routine (see (14)) whenever a fighter is given the scramble order.

(5) This determines, via subroutine COLLISION, an estimated interception course; if there is such a course, it calculates the expected time to interception, TINT (excl. take-off delay), and the coordinates of the interception point (PX,PY) relative to the fighter base.

(6) This is calculated, for each fighter type, CAP and fighter base, just once during a model run, in subroutine INPUT. A test is also made that the fighter has sufficient fuel to fly to CAP, with a fuel reserve for combat.

(7) JBASE is the serial of the base providing the fighter which can carry out its mission in the shortest total time TMIN1. If the fighters are scrambled to intercept a raid then (XRELMIN, YRELMIN) denote the coordinates of the expected interception point relative to base JBASE.

(8) If TMIN1 still equals 10^6 no suitable fighters are available at any of the fighter bases. The number of fighters allocated at this entry to subroutine FCHOOSE is calculated (=K-1), this

information is output and control returns to the calling subroutine.

(9) If the fighter is assigned to cruise to a CAP its serial IF is determined and its control mode MODEC(IF) set equal to 1 by calling subroutine FCREA(IF,1). Similarly, if it is to intercept a raid and a Close Control (CC) facility is available (ICC=1), together with an intercept controller (NGC>0), subroutine FCREA is called with parameter list (IF,3); this derives the fighter serial IF and sets the control variable MODEC(IF)=3. Otherwise, if the fighter is to intercept a raid but CC is not available the fighter is assumed to adopt its course under Broadcast Control (BC), which corresponds to MODEC=2, so that subroutine FCREA is called with parameter list (IF,2).

10. The parameters peculiar to a fighter flying to a CAP are as follows:

(i) NEXTCAPPT(IF)= serial of the next CAP point towards which the fighter is flying.

(ii) KCAP(IF) stores the serial ICAP of the fighter CAP.

(iii) Profile 1 is currently the 'cruise' profile, so KPROF(IF)=1.

(iv) The fighter's initial vector to the CAP must be set up; this is represented by the x- and y- components of its velocity and acceleration, (FVX(IF),FVY(IF)) and (FFX(IF),FFY(IF)) respectively.

11. This sets up the arrival of the fighter on CAP, an event type 9, in time TTOT from now. The address of the pointer to the event data in the event list is stored in the variable LOCEVF2(IF), in case the event must later be cancelled.

12. The parameters peculiar to a fighter scrambled to intercept raid IR are as follows:

(i) The number of fighters attacking the raid must be increased by one.

(ii) IF must be inserted at the end of the list of fighters attacking raid IR, ie the IATTACK array.

(iii) ITARGET(IF) stores the serial IR of the raid which fighter IF is attacking.

(iv) Profile 2 is currently the interception profile, so KPROF(IF)=2.

13. This sets up the expected interception (event type 12) in time TINT= TTOT(JBASE) from now. It also sets up the fighter's initial vector, ie (FVX(IF),FVY(IF)),(FFX(IF),FFY(IF)), for which it needs the coordinates (XRELMIN,YRELMIN) of the expected interception point relative to the base.

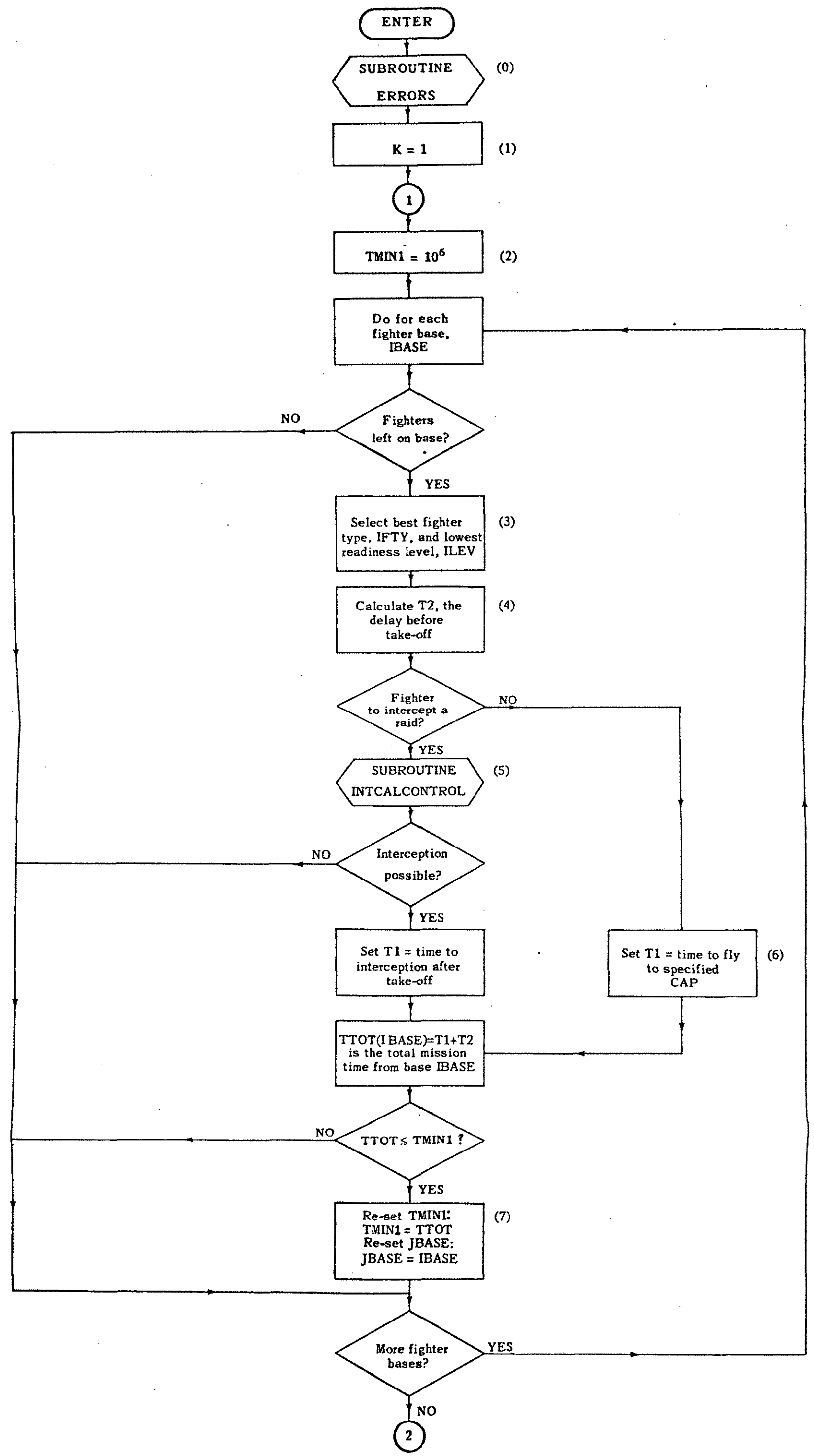


FIGURE 16.1. PROGRAM LOGIC – SUBROUTINE FCHOOSE (PART (i))

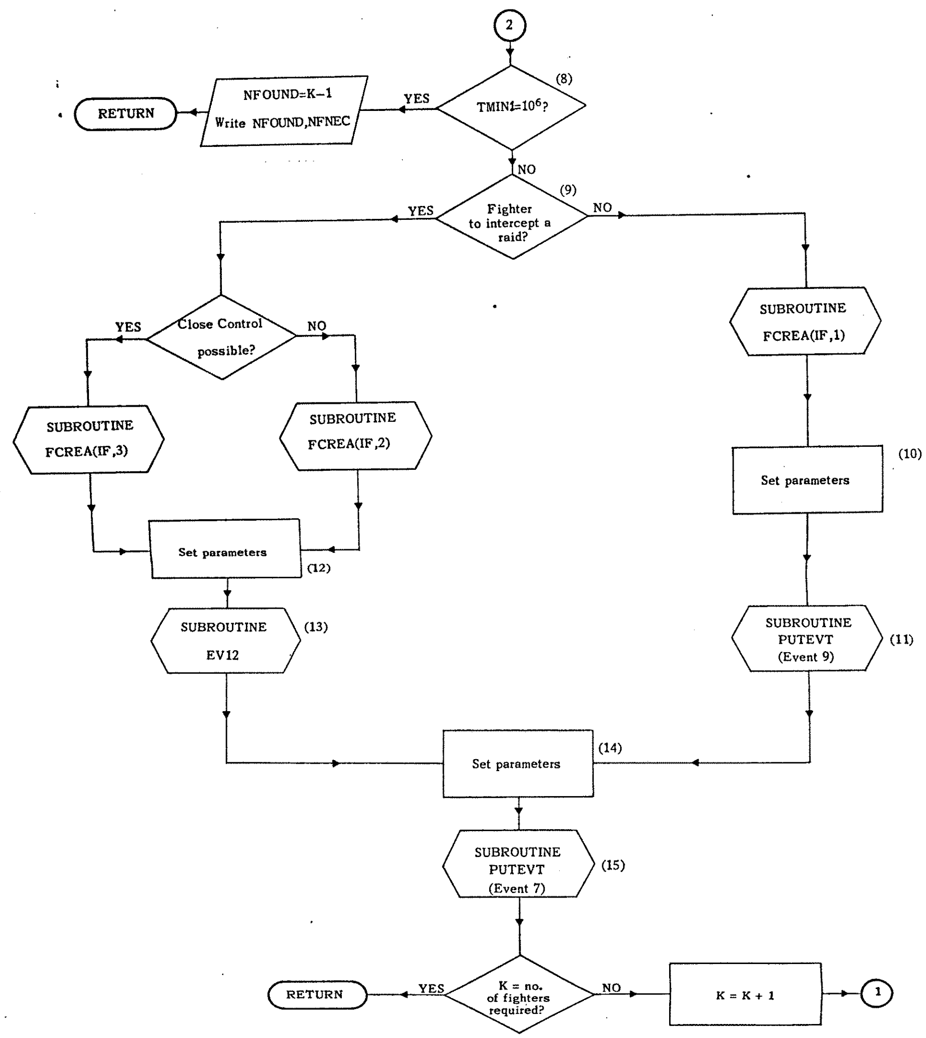


FIGURE 16.2. PROGRAM LOGIC – SUBROUTINE FCHOOSE (PART (ii))

(14) The parameters common to fighters assigned to CAP or to intercept a raid are as follows:

(i) Decrease the number of fighters available at the base (NRLEV,NFCUM).

(ii) Increase the queue for take-off at the base (NQ,TBGO).

(iii) Set MODER(IF)=0 to correspond to a fighter between allocation and take-off.

(iv) JPPT(IF) stores the serial of the last profile point reached and is here set equal to 1.

(v) Store the fighter type, its base and readiness level, its initial fuel load and consumption rate, its initial x- and y-coordinates (=coordinates of the base) and its initial missile fit.

(15) This sets up the fighter take-off (an event type 7) in time T2=TONBASE(JBASE) from now (see (4)).

17. SUBROUTINE FCREA(IF,MODE)

This subroutine creates a fighter with control mode MODEC(IF) equal to MODE. The number of active fighters is increased by one and the serial JF of the first fighter with a control mode greater than MODE is determined. (IFM(I), if non-zero, is the serial of the first fighter in mode I, for I=1,2,3,4). If there are no such fighters with a control mode MODEC greater than MODE, then IFM(I)=0,I=MODE+1,...,4. In this case (and in particular if MODE=4) JF is set equal to IFM(5). This is the next fighter serial available for use and is never zero. (IFM(I), I=1,..,5,are initialised at the beginning of each replication in subroutine REPINIT).

IF is then set equal to the next available serial, IFM(5); this must then be updated so that it continues to store the next available serial by setting:

IFM(5)=IUP(IFM(5)) (17.1)

The required value of MODEC(IF) is then defined:

MODEC(IF)=MODE (17.2)

It is necessary to change the pointers of IF and JF, so that IF is positioned at the end of the list of fighters in mode MODE, and points to the first fighter with control mode greater than MODE (or the first available fighter serial, if there are no such fighters). This is achieved by a call to subroutine FSEQCH, with the parameter list (IF,JF). Finally, if IF is the only fighter in mode MODE (so that, currently, IFM(MODE)=0) then IFM(MODE) is reset:

IFM(MODE)=IF (17.3)

18. SUBROUTINE FDESTR(IF)

This subroutine removes fighter IF from the list of active fighters and is called from subroutines FDESTR1 and FCHMODE.

The number of fighters currently active, NF, is decreased by one and, if IF is currently under Close Control the number of available intercept controllers is increased by one. The serial JF of the fighter immediately above IF in the list is obtained from JF=IUP(IF). IF is removed from its current position in the active fighter list and placed below IFM(5) (the first fighter serial available for allocation) by a call to subroutine FSEQCH with parameter list (IF,IFM(5)). The serial IF becomes available for re-use if another fighter is now allocated and scrambled-in fact it becomes the first available serial - by setting

$$IFM(5)=IF \qquad (18.1)$$

If IF was not the first fighter in mode MODE=MODEC(IF), so that $IFM(MODE)\neq IF$, then control returns to the calling subroutine at this point. Otherwise, IFM(MODE) must be reset; if IF was the only fighter in mode MODE, so that $MODEC(JF)>MODE$, then IFM(MODE) is set equal to zero, otherwise it is set equal to JF.

19. SUBROUTINE FDESTR1(IF)

This routine removes fighter IF from all processing when it can no longer contribute to the defence system. It may be called from subroutines REATTACK if the fighter has run out of missiles or EV9 if, after a cancelled interception, it has insufficient fuel to fly to a CAP or a base. It may also be called at an event type 12 if the measure of fighter effectiveness is defined to be simply interception or collision (event type 12) rather than missile-splash (event type 16).

Subroutine DELETE is called with parameters LOCEVF1(IF),LOCEVF2(IF) and LOCEVF3(IF) to eliminate all possible events pending for this fighter. Subroutine FDESTR(IF) then removes fighter IF from the list of active fighters; the serial IF then becomes available for reallocation.

20. SUBROUTINE FFIND(NFSHORT,NFSHORT1)

This subroutine is called to find a specified number NFSHORT of fighters which are already allocated, to intercept a specified raid IR. Only fighters with control modes MODEC=1 or 2 are considered, ie fighters cruising to or on CAP, or under simple Broadcast Control. All necessary changes in fighter parameters are carried out for those fighters found by this subroutine and control returns to the calling routine with the variable NFSHORT1, the shortfall in the number of fighters required ($NFSHORT1 \geq 0$). If necessary, subroutine FCHOOSE may then be called to attempt to meet this remaining shortfall from the unallocated fighters on alert at the fighter bases. Throughout, the emphasis is on least-time-to-intercept, so that those eligible fighters are chosen which it is believed can intercept the raid most quickly.

Figures 20.1 and 20.2 illustrate the logic of this subroutine. Subroutine ERRORS is first called; this ensures that in the subsequent calculation of interception courses (via subroutine INTCALCONTROL) the same errors in the estimates of raid position, speed and heading apply to all fighters. NREQ is the total number of fighters required to attack the raid, ie

$$NREQ=NATTACK(IR)+NFSHORT \quad (20.1)$$

TINT1(I), I=1,.., NFSHORT will contain, in order of increasing time, the interception times of the NFSHORT fighters which can intercept the raid most quickly; this array must first be initialised. Subroutine SETIF is called with parameter list (1,2), to correspond to a search through fighter control modes MODEC=1 and 2 only. Each such fighter is then processed in turn.

The variable INIT is first calculated; this is the nearest point on profile 2, the interception profile, with a corresponding speed less than or equal to the current fighter speed. Subroutine INTCALCONTROL then determines if an interception by this fighter is believed to be possible. If it is (so that IMPOSS=0), the number of fighters attacking the raid, NATTACK(IR), is increased (unless NATTACK(IR) already equals NREQ). Five arrays-TINT1, PX1,PY1,IFMIN,INIT1 - store the interception data for each fighter considered. This comprises, respectively, the time to intercept, x- and y- coordinates of the expected interception point (relative to the current fighter position), fighter serial and the profile-point serial, INIT. This is stored in order of increasing time to interception, ie

$$TINT1(I) \leq TINT1(L) \quad , \quad 1 \leq I \leq L \leq NFSHORT \quad (20.2)$$

If

$$TINT<TINT1(NFSHORT) \quad (20.3)$$

the interception information for this fighter is inserted into the appropriate place in the arrays. This procedure is repeated for each fighter with MODEC=1 or 2.

As shown in Figure 20.2 the number of fighters found by this routine, NFOUND, and the number still required (if any), NFSHORT1, are calculated and printed. If NFOUND>0, each fighter IF assigned to intercept the raid has the appropriate parameters altered. Thus subroutine FCHMODE sets MODEC(IF)=2 if the fighter adopts Broadcast Control and MODEC(IF)=3 if it is under either Close Control or Date Link Control. Similarly the other control variable MODER(IF) is set equal to 5,11 or 12 - see Figure 0.5. (If the fighter is not yet airborne, MODER(IF) is instead set at the fighter take-off - event type 7). The serial IR of IF's target raid is stored in ITARGET(IF) and correspondingly the serial IF is inserted into the list of fighters attacking raid IR, in the IATTACK array. Finally, the interception data is retrieved from the arrays described above in order to set up the interception event (event type 12) in subroutine EV12 and, if the fighter has taken-off (ie MODER(IF)> 0), the time of the next profile-point in profile 2 (event type 8) in subroutine EV8.

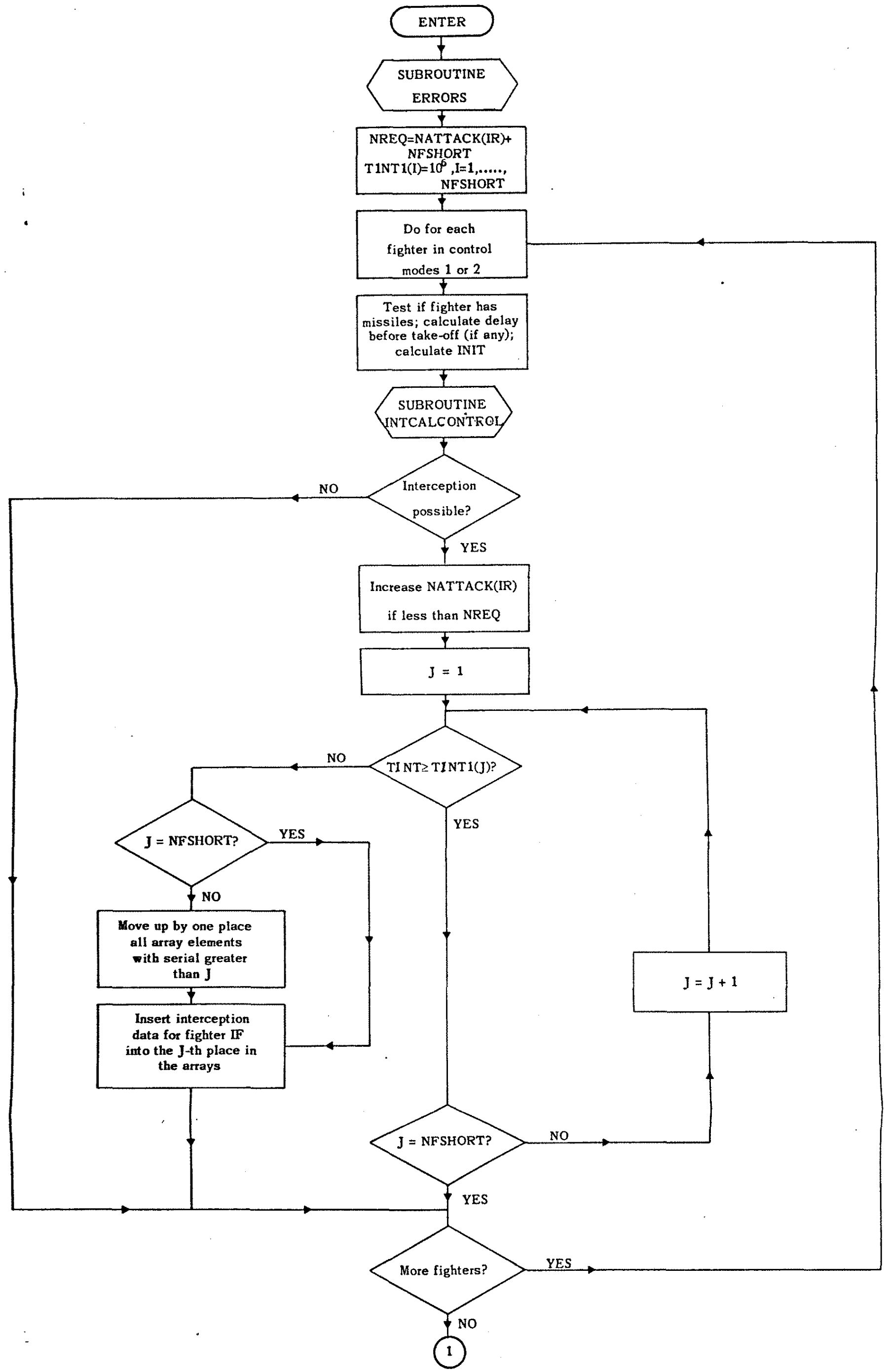


FIGURE 20.1. PROGRAM LOGIC – SUBROUTINE FFIND (PART (i))

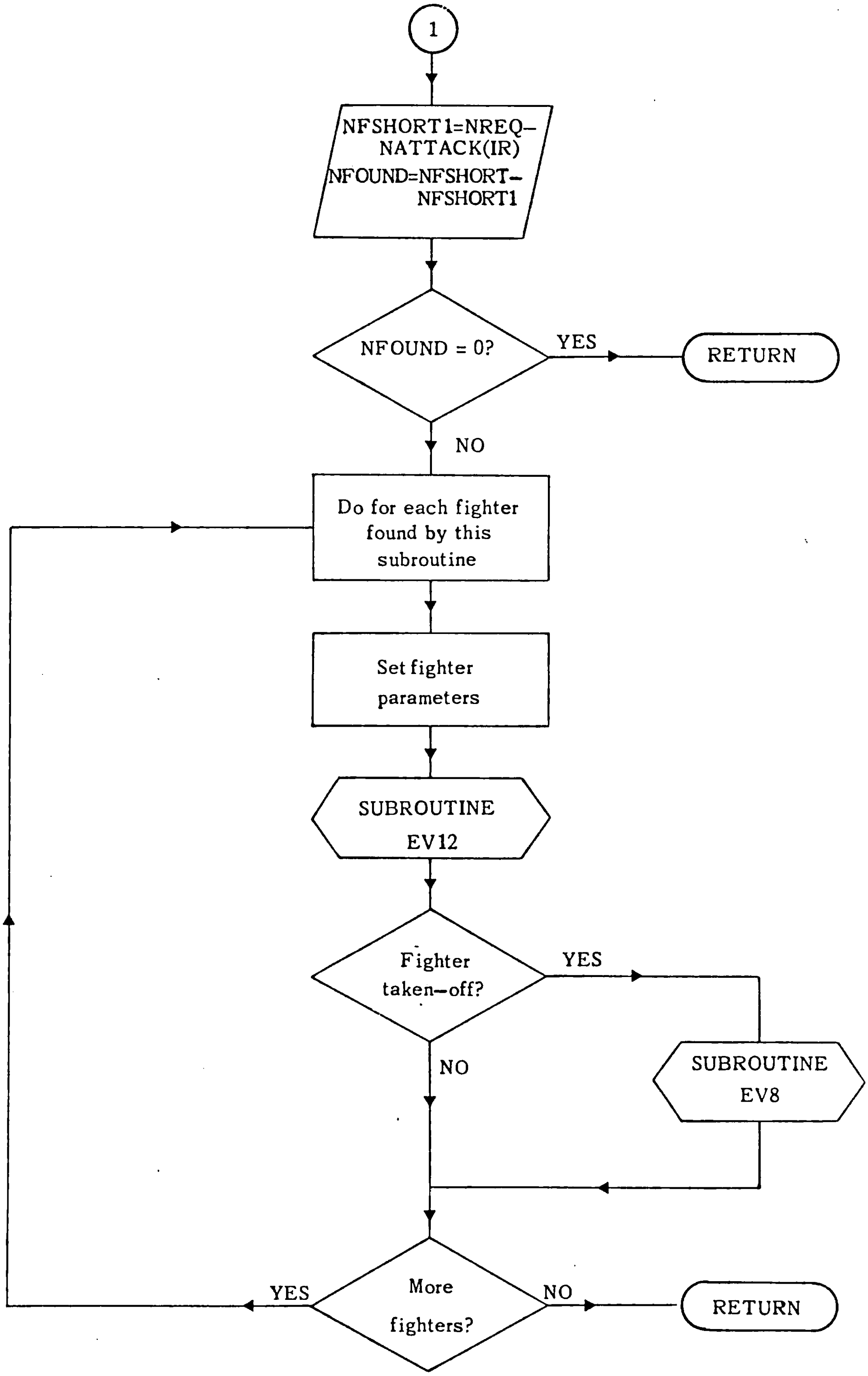


FIGURE 20.2. PROGRAM LOGIC – SUBROUTINE FFIND (PART (ii))

21. SUBROUTINE FIGHTERTURN

This routine is called from subroutine MOVE1F to update fighter IF's position and velocity during a turn between attacks on raid IR. It contains no processing, since as a first approximation it is currently assumed that the fighter's position is 'frozen' during such a turn, which is made with constant angular velocity. In any extension to the reattack sequence in the Fighter Model, it would be straightforward to incorporate in this routine the fighter's spatial movement during a turn.

22. SUBROUTINE FSEQCH(IF,JF)

This subroutine moves fighter IF to the position immediately below fighter JF in the fighter list, and is called from subroutines FCREA and FDESTR. If IF=JF or IF=IDOWN(JF) no action is necessary, otherwise the IUP and IDOWN pointers are adjusted as shown in Figures 22.1 and 22.2. IF is removed from its original position and the list closed around it, it is inserted between the original IDOWN(JF) and JF and the pointers adjusted accordingly.

23. SUBROUTINE GAINREAD(GAIN,ICONV)

This subroutine reads, and stores in the appropriate array, data in the form of a polar diagram. It is called from subroutines INPUT and MISSREAD and currently is used to input the following information.

(i) Radar Data

GR1(46,3) AI Radar Antenna Gain Pattern (up to 3 AI radar types)

TXSEC(46,2) Enemy Aircraft Radar Cross-Sections (2 types of enemy aircraft-bombers/SSJs and specialist jammers)

(ii) Jammer Data

GJR(46,2) Jammer Receiver Polar Diagram (2 types-bomber/SSJ and specialist jammer)

GJT(46,2) Jammer Transmitter Polar Diagram (2 types-bomber/SSJ and specialist jammer)

(iii) Missile Data

RMAX(46,2) Outer Boundary of Missile Launch Success Zone (2 types of missile)

RMIN(26,2) Inner Boundary of Missile Launch Success Zone (2 types of missile)

TOF(46,2) Time of Flight from launch at outer boundary to splash

The parameter list of this subroutine is (GAIN,ICONV).

(i) GAIN denotes the array (ie one of those listed above) in which

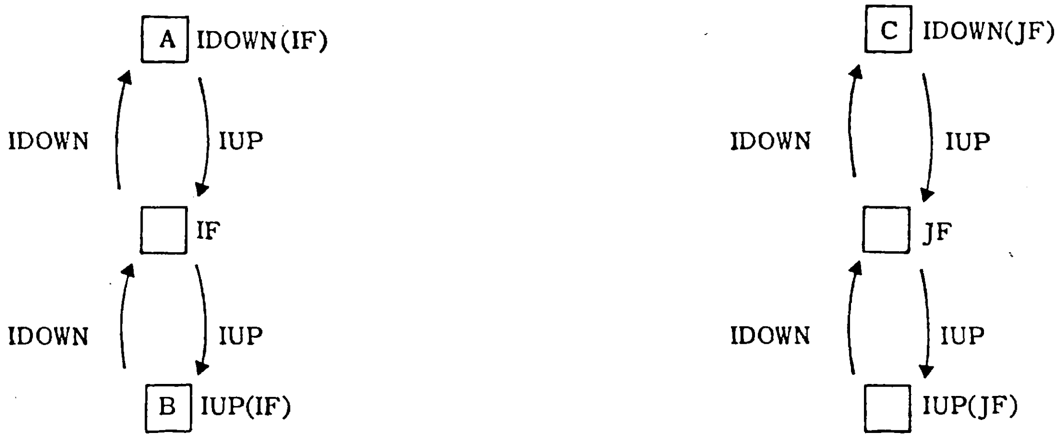


FIGURE 22.1. ORIGINAL POSITIONS OF IF AND JF – SUBROUTINE FSEQCH

FIGURE 22.2. NEW POSITIONS OF IF AND JF – SUBROUTINE FSEQCH

Part 2 - p.74.

Description of subroutine GAUSSRV(R) should read:
This routine is called from subroutines ERRORS and ROTATE to sample a random variable from a normal distribution with mean 0 and s.d. 1. It simply adds a number of i. i. d samples from a uniform distribution.

the polar diagram data is to be stored. All polar diagram patterns are assumed to be symmetrical about a central axis, so that gain patterns, aircraft cross-sections, etc need only be determined as functions of angles from 0^{o} to 180^{o}. Currently, all polar diagram patterns are stored in arrays of dimension 46, so that this information is determined at all multipes of 4^{o} (=180/(46-1)) from 0^{o} to 180^{o}. If greater accuracy is required in the specification of gain patterns the dimensions of the arrays listed above may simply be increased accordingly.

(ii) <u>ICONV</u> is a conversion factor; if ICONV=1 the polar data (normally AI radar antenna gain patterns and jammer receiver and transmitter polar diagrams) is converted from dB to absolute, while if ICONV=2 it is multiplied by 1000 (the maximum and minimum missile launch ranges are currently input in km).

For any call to subroutine GAINREAD, the polar data, and the angles at which the polar data are specified, are read into the arrays GVALUE and GANGLE respectively. It is assumed that GANGLE(1)=0, and the final value of GANGLE is 180^{o}, otherwise the angles at which GVALUE points are specified are quite arbitrary. There is no relationship assumed between the number of input values of GANGLE and the number of equally spaced angles (of 4^{o}) at which the polar diagrams are calculated. The GANGLE and GVALUE arrays are currently of dimension 20; if more points are required to be specified in a gain pattern, this dimension is simply increased accordingly. (Note the fairly rigid input format used for this data, namely 12F6.0). The values of GAIN(K), K=1,...., 46, where the integer K corresponds to the angle 4(K-1) degrees, are then determined by linear interpolation of the values of GVALUE, specified at the angles GANGLE.

24. SUBROUTINE GAUSSRV(R)

This routine is called from subroutines ERRORS and ROTATE to produce an approximately normal random variable, R, with mean 0 and standard deviation 1. It simply adds a sufficient number of identical random variables, each of which is uniformly distributed between 0 and 1.

25. SUBROUTINE GETEVT(EVTIME,IEVTYP,IUNIT)

This is a standard event-processing subroutine, as described in Lambeth (1978). Briefly, it retrieves the contents of the next event (time=EVTIME, type=IEVTYP, raid or fighter serial=IUNIT) from the EVENT and IEVENT arrays, via the pointer at the head of the event queue. The emptied cell is returned to the stack of free cells and the pointers NXTEVT to the next event to be processed and NXTFRE to the first cell of the linked list of free cells are updated.

26. FUNCTION IANG(ANGLE)

This function is called from subroutines COLLISION, JPOWER,LAUNCHTEST,RADAR and SOJPOWER, with the angle ANGLE in radians specified in the parameter list. It determines the nearest integer angle to ANGLE at which polar diagrams are specified; these are currently defined over the range 0°-180° at 4° intervals (see subroutine GAINREAD). The integer IANG ($1 \leq IANG \leq 46$) is determined as follows; the required angle, in degrees, is then simply 4.(IANG-1).

ANGLE is first converted to degrees:

$$ANGDEG=ANGLE.(180/PI) \quad (26.1)$$

IANG is then found from:

$$ANGDEG=(IANG-1).4 + REM \quad (0\leq REM<4) \quad (1\leq IANG\leq 46) \quad (26.2)$$

IF REM$>$2, then

$$IANG=IANG+1 \quad (1\leq IANG\leq 46) \quad (26.3)$$

27. SUBROUTINE IATTDEL(I)

I denotes the position of fighter IF in the list of fighters attacking raid IR, ie the array (IATTACK(IR,K), K=1,.....,NATTACK(IR)), where NATTACK(IR) is the total number of fighters attacking the raid. This routine deletes IF from this list as follows:

$$IATTACK(IR,K)=IATTACK(IR,K+1) \quad (K=I,.....,NATTACK(IR)-1) \quad (27.1)$$

$$NATTACK(IR) = NATTACK(IR)-1 \quad (27.2)$$

28. SUBROUTINE IATTFIND(I)

This subroutine is called, for a fighter IF attacking a raid IR, to find the position, I, of IF in the IATTACK array of IR. This array is the (ordered) list of all fighters which are attacking raid IR, so that IF and IR are related by

$$IF=IATTACK(IR,I) \quad (28.1)$$

29. LOGICAL FUNCTION IDFUEL(XT,YT,V,FL,DT)

This logical function determines whether a fighter with initial speed V and fuel FL can reach the point (XT,YT) on an interception profile in flight time DT, with sufficient fuel to engage in combat and then return to a fighter base safely. If so, IDFUEL is set equal to TRUE, and if not it is set equal to FALSE.

Function IDFUEL is called only from subroutine INTCALCONTROL. (XT,YT) denotes the calculated interception point, while INIT+1 is the next point on profile 2 (the current interception profile) for the fighter. Subroutine SFROMT calculates the fuel FLT used in time DT, ie the fuel used by the fighter in flying (in profile 2) to the expected interception point (XT,YT). FAVAIL is then set equal to the expected fuel available after reaching (XT,YT) and allowing for a fuel reserve FRES for combat (FRES is a function only of the fighter type IFTY). Each fighter base is then checked to determine whether the fighter can cruise from (XT,YT) to the base with fuel FAVAIL. (The fighter's cruise speed and cruise fuel consumption rate are taken to be those corresponding to the final points on profile 1).

30. SUBROUTINE INITVT

This is a standard event processing subroutine - see Lambeth (1978). It is called from subroutine REPINIT at the beginning of each replication to initialise the event system. It sets up the stack of empty event cells (IEVENT(1, ·)) and gives initial values to the next-event (NXTEVT) and next-free-cell (NXTFRE) pointers.

31. SUBROUTINE INPUT

Subroutine INPUT is called just once, from the Master Segment MASTER DFX. It reads in all the program data, calling subroutine GAINREAD to read AI radar and jammer gain patterns and enemy aircraft cross-section data, and subroutine MISSREAD to read all missile data. Subroutines RREAD and IREAD are repeatedly called to read one-dimensional real and integer arrays respectively. The definition of all program arrays and variables, including input data arrays and variables, is given in the Glossary. Note that data items are input in the units most commonly used for those items, as shown in the illustrative data file (Annex B). All data items are then converted to the internal program units - radians, metres, seconds, pounds (fuel), watts (radar and jammer threshold detection levels, powers, losses and gains) and MHz (radar and jammer carrier frequencies and bandwidths; the only exception is a linear ranging modulation of the AI carrier frequency, assumed expressed in units of Hz/sec). In particular powers, gains etc. are input in dB, and converted to absolute units (ie watts) by Function DBTOABS.

Subroutine INPUT also initialises all variables which are independent of the replication machinery. This includes all statistical variables, which are initialised in subroutine STATSINIT. Subroutine RTRACKS sets up the track parameters for each enemy raid. Logical Function INTERSECT calculates, if they exist, the points of intersection of the raid tracks with the three warning lines. The corresponding detection events (event types 3,4 and 5) for each raid are then set up at the beginning of each replication in subroutine REPINIT.

The following radar equation constants are calculated once only during a model run, for each fighter type:

> CPATR (Clutter Power at Radar), RPATJ (Radar Power at Jammer), PJATR (Power of Jammer at Radar), RPATR (Radar Power returned at Radar).

Finally, the variables PE,QE,RE and SE are initially set equal to 1.0. In deterministic mode they remain equal to 1.0, while in stochastic mode they take values sampled from a N(0,1) distribution, for use in calculating interception courses with errors in the estimates of raid position, speed or heading.

32. SUBROUTINE INTCALCONTROL(TDELAY,ICALL)

Subroutine INTCALCONTROL prepares data for the interception algorithm, subroutine COLLISION, in the calculation of interception courses. On returning to subroutine INTCALCONTROL from subroutine COLLISION details of the interception calculations are printed and, if an interception is possible, further tests on fighter fuel state and crossing of out-of-bounds regions are carried out. The parameter ICALL identifies the calling routine of subroutine INTCALCONTROL and TDELAY the delay before the fighter can adopt an interception course, as follows:

(i) ICALL=1 when this routine is called from subroutine EVENTS, at event type 4. This corresponds to the recalculation of interception courses for each fighter IF which has already detected and is attacking raid IR under autonomous control, when Ground Control first acquires track information on the raid. It is assumed that these fighters immediately adopt their new interception courses, so that TDELAY=0.0.

(ii) ICALL=2 when called from subroutine EVENTS at event type 14, to recalculate fighter IF's interception course after a track-change by its target, raid IR. TDELAY=0.0 unless the fighter has not yet taken-off, in which case it is set equal to the delay before take-off.

(iii) ICALL=3 when this routine is called from subroutine FCHOOSE. This calculates whether an unallocated fighter of type IFTY at base IBASE can intercept raid IR. TDELAY is set equal to the delay before take-off, calculated as usual in the calling routine. This particular call merits attention because the necessary fighter variables - initial position, speed, fuel state, etc - do not refer to a fighter with a well-defined serial.

(iv) ICALL=4 when this routine is called from subroutine FFIND to determine whether fighter IF, currently allocated to cruise to or on CAP, can intercept raid IR. Again, TDELAY=0.0 unless the fighter has not yet taken-off.

(v) ICALL=5 when called from subroutine RADEVENTS to calculate an interception course for fighter IF when it first achieves burnthrough, on its AI radar, against raid IR. It is assumed that the fighter can adopt its new course immediately so that TDELAY=0.0.

(vi) ICALL=6 when this routine is called from the first section of subroutine REATTACK. This is a preliminary calculation, to find the most likely next target for fighter IF in the reattack sequence. It is assumed here that the fighter can instantly turn onto its new heading after its previous attack, so that TDELAY=0.0.

(vii) ICALL=7 when this routine is called from the second section of subroutine REATTACK. Having found the most likely next target for the fighter its interception course is calculated more precisely, taking into account an approximation to the fighter's turn-time, denoted by TTURN1; this is calculated at the previous entry to subroutines INTCALCONTROL and COLLISION, with ICALL=6. Thus, in this subroutine call TDELAY=TTURN1.

The first calculations performed in this routine simply set up the following variables:

(RXD,RYD): current raid position

(X,Y) : current fighter position

FUELORIG : current fighter fuel state

VØ : current fighter speed

INIT : the nearest point on profile 2, the interception profile, with a corresponding speed less than or equal to the current fighter speed.

Note that, if ICALL≠3, subroutine RELCOORDS calculates the raid coordinates (RXD,RYD) while subroutine MOVE1F updates the fighter position. If ICALL=3 then INIT=1, while if ICALL=4 INIT is calculated in the calling routine, FFIND. Otherwise the fighter has already adopted an interception profile, currently taken to be profile 2, so that INIT=JPPT(IF), the serial of the last profile-point reached.

The estimated raid coordinates, speed and heading are then derived from the actual values of these parameters and the (deterministic or stochastic) errors in the estimates of these parameters. The standard deviations of the errors in the estimates of the raid parameters, for fighter or GC information, are specified in the input data. If the model is in deterministic mode (MODERUN=0) the variables PE,QE,RE and SE are all set equal to 1.0 in subroutine INPUT. In stochastic mode these variables all take values - set in the calling routine - sampled from a N(0,1) distribution (via subroutine ERRORS). The current standard deviations of the errors in the estimates of raid position, speed and heading are, respectively, RPOSERR(IR),RSPEEDERR(IR) and RDIRERR(IR). The magnitudes A,B,C,D of the errors in the estimates of the raids x- and y- coordinates, speed and heading are then given by:

A=RPOSERR(IR).PE

B=RPOSERR(IR).QE

C=RSPEEDERR(IR).RE (32.1)

D=RDIRERR(IR).SE

The estimated raid speed, RVT, is then

RVT=RV(IR)+C (32.2)

If θ_t is the actual raid heading, the estimated raid heading θ_f is

$$\theta_f = \theta_t + D \qquad (32.3)$$

The parameters actually used in later calculations are the estimated x- and y- components of the raid velocity, denoted by RAIDVX and RAIDVY respectively. These are given by

$$RAIDVX = RVT.\cos\theta_F$$

$$= RVT.\left\{\frac{RVX(IR)}{RV(IR)}\cos D - \frac{RVY(IR)}{RV(IR)}\sin D\right\} \qquad (32.4)$$

Similarly

$$RAIDVY = RVT.\left\{\frac{RVY(IR)}{RV(IR)}.\cos D + \frac{RVX(IR)}{RV(IR)}\sin D\right\} \qquad (32.5)$$

Finally the estimated raid coordinates relative to the fighter after the delay TDELAY are:

RELX =(RXD+A)-X+(RAIDVX.TDELAY)

RELY =(RYD+B)-Y+(RAIDVY.TDELAY) (32.6)

where (RXD,RYD) are the actual raid coordinates and (X,Y) are the fighter coordinates.

Subroutine COLLISION then determines if the fighter can intercept the raid on its estimated track. If not, on return from subroutine COLLISION the control parameter IMPOSS equals 1 and control returns immediately to the calling routine. Otherwise subroutine COLLISION sets IMPOSS=0 and returns the coordinates (PX,PY) of the expected interception point relative to the fighter and the time TINT to interception (excluding the specified delay TDELAY).

The time for the fighter to turn onto its interception course is only taken into consideration when the fighter is in the reattack sequence. Otherwise, the raid is assumed to be sufficiently far from the fighter for the fighter's turn-time to be neglected. Thus if ICALL$\leqslant$5, so that the fighter under consideration is not in the reattack sequence, information on the interception course is printed, while subroutine OUTOFB checks that this proposed course does not take the fighter into one of the specified out-of-bounds regions. Subroutine IDFUEL then checks that after reaching the expected interception point and taking into account a combat fuel allowance (specified in the input data), the fighter will still possess sufficient fuel to return to one of the fighter bases. If the proposed interception course does not go out of bounds or consume excessive fuel and if ICALL=1,2 or 5, subroutine EV12 sets up the new course, after which control returns to the calling routine. If ICALL=3 or 4, so that this routine is called from subroutine FCHOOSE or subroutine FFIND, this may be just one of a number of fighters being tested, to determine which can intercept the raid most quickly. In these cases the interception data is returned to the calling routine which, after selecting the fighters to carry out the planned interception, calls EV12 directly to set up their new courses.

If the fighter is in the reattack sequence, so that ICALL=6 or 7, the angle α (degrees) between the fighter's current heading and its proposed heading is calculated in subroutine DOTPROD (see Figure 32.1). As a first approximation the fighter is assumed to be capable of making this turn in a time TTURN seconds at a turn rate of 3.0*TRNRATE degrees/second where TRNRATE is specified in the input data. Thus

$$\text{TTURN}= \frac{|\alpha|}{3.0*\text{TRNRATE}} \qquad (32.7)$$

The turn-time TTURN is added onto the interception time TINT and the processing described above - the printing of interception information and the calls to subroutines OUTOFB and IDFUEL - continues. (Note that the fuel consumed during the turn by the fighter is neglected). If ICALL=6, so that the most likely next flight for the fighter to attack within this raid is being chosen, control then returns immediately to subroutine REATTACK. If ICALL=7, so that the target flight has been chosen and a more accurate interception course is being calculated, and if this course does not go out-of-bounds or consume excessive fuel, subroutine EV12 finally sets up this new course.

33. LOGICAL FUNCTION INTERSECT(R1X,R1Y,R2X,R2Y,S1X,S1Y,S2X,S2Y,A)

This function is called from subroutine INPUT for each raid and each warning line to determine if the raid track-leg from the point (R1X,R1Y) to the point (R2X,R2Y) intersects the warning line segment from the point (S1X,S1Y) to the point (S2X,S2Y), using a simple application of Cromer's Rule. If there is no intersection control returns with INTERSECT=FALSE, otherwise INTERSECT=TRUE and the variable A ($0 \leq A \leq 1$), defined in Figure 33.1, is returned to subroutine INPUT. Knowing the times at which the raid is at the points (R1X,R1Y) and(R2X,R2Y), the time at which it crosses the warning line may then be calculated.

34. SUBROUTINE IREAD(L,N)

This subroutine is called from subroutine INPUT to read an integer array, L, of dimension N.

35. SUBROUTINE ITERATE(T,A,B,C,D,E,F,*,*,T1)

This routine is called from subroutine COLLISION to carry out a Newton-Raphson iteration to find a root of a quartic equation. In the notation of subroutine COLLISION, the elements in the parameter list are as follows:

$$A = R2\emptyset = |\underline{R}_0 + \underline{v}t_i|^2$$

$$B = 2RV\emptyset = 2\,(\underline{R}_0 + \underline{v}t_i).\underline{v}$$

$$C = V2 = |\underline{v}|^2 \qquad (35.1)$$

$$D = ST\emptyset = s_i$$

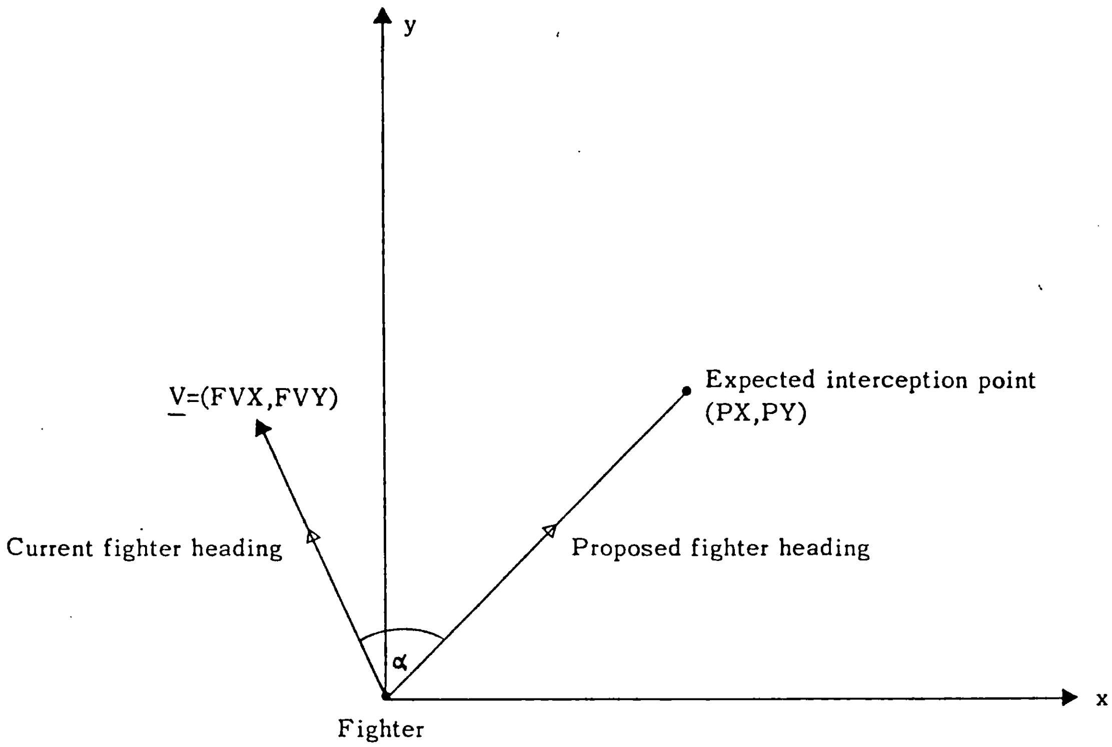


FIGURE 32.1. FIGHTER ANGLE OF TURN – SUBROUTINE INTCALCONTROL

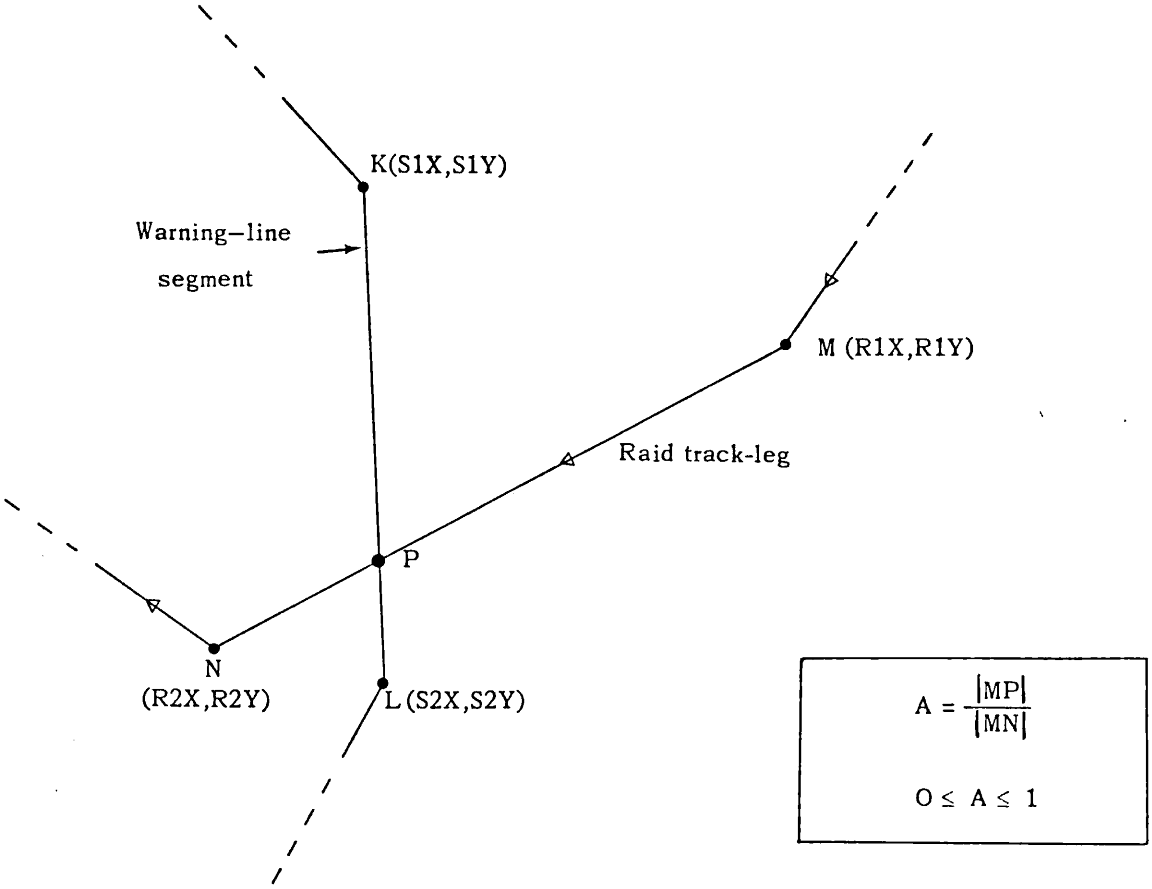


FIGURE 33.1. INTERSECTION OF RAID TRACK-LEG AND WARNING LINE SEGMENT – LOGICAL FUNCTION INTERSECT

$$E = V\phi = V_i$$
$$F = \frac{1}{2} ACC = \frac{1}{2} f_i \qquad (35.1)$$

Control returns to the second specified label if a root is found, otherwise it returns to label 1. The parameter T is the specified initial approximation to the root; each successive approximation is compared with the specified upper bound T1.

The equations in subroutine ITERATE give:

$$P1 = E + F.T = V_i + \frac{1}{2} f_i T$$
$$P2 = B + C.T = 2(\underline{R}_0 + \underline{v} t_i).\underline{v} + |\underline{v}|^2 T \qquad (35.2)$$
$$P3 = D + P1.T = s_i + V_i T + \frac{1}{2} f_i T^2$$

The quartic polynomial, denoted by FUNC, is then given by equation (3.7):

$$FUNC = A + P2.T - P3.P3 \qquad (35.3)$$
$$= |\underline{R}_0 + \underline{v} t_i|^2 + 2(\underline{R}_0 + \underline{v} t_i).\underline{v} T + |\underline{v}|^2 T^2 - \left(s_i + V_i T + \frac{1}{2} f_i T^2\right)^2 \qquad (35.4)$$

The slope of the quartic at the approximation T is denoted by SLOPE, so that successive approximations are given by

$$T = T - \frac{FUNC}{SLOPE} \quad ; \qquad (35.5)$$

Convergence is assumed when

$$|FUNC| \leq \varepsilon \text{ , where currently } \varepsilon = 100 \text{ (metre)}^2 \qquad (35.6)$$

The iteration is assumed to be diverging if $T > T1$, while from subroutine COLLISION it is necessary that $SLOPE < 0$.

36. SUBROUTINE JDETLOGIC(JDET)

This routine is called from subroutine RADAR and determines the type of jamming emitted by jammer type IJTYPE in raid IJ against fighter IF. This information is returned in the parameter list in the variable JDET, which may take the values 0,1 or 2:

<u>JDET=0</u> corresponds to the fighter's radar remaining undetected by the jammer, and therefore unjammed (unless by accident).

<u>JDET=1</u> corresponds to the fighter's radar continuously detected by the jammer and continuously jammed. (It is assumed that the frequency range which the jammer can detect equals the frequency range which it can jam although the jammer receiver and transmitter may have different gain patterns specified in the input data).

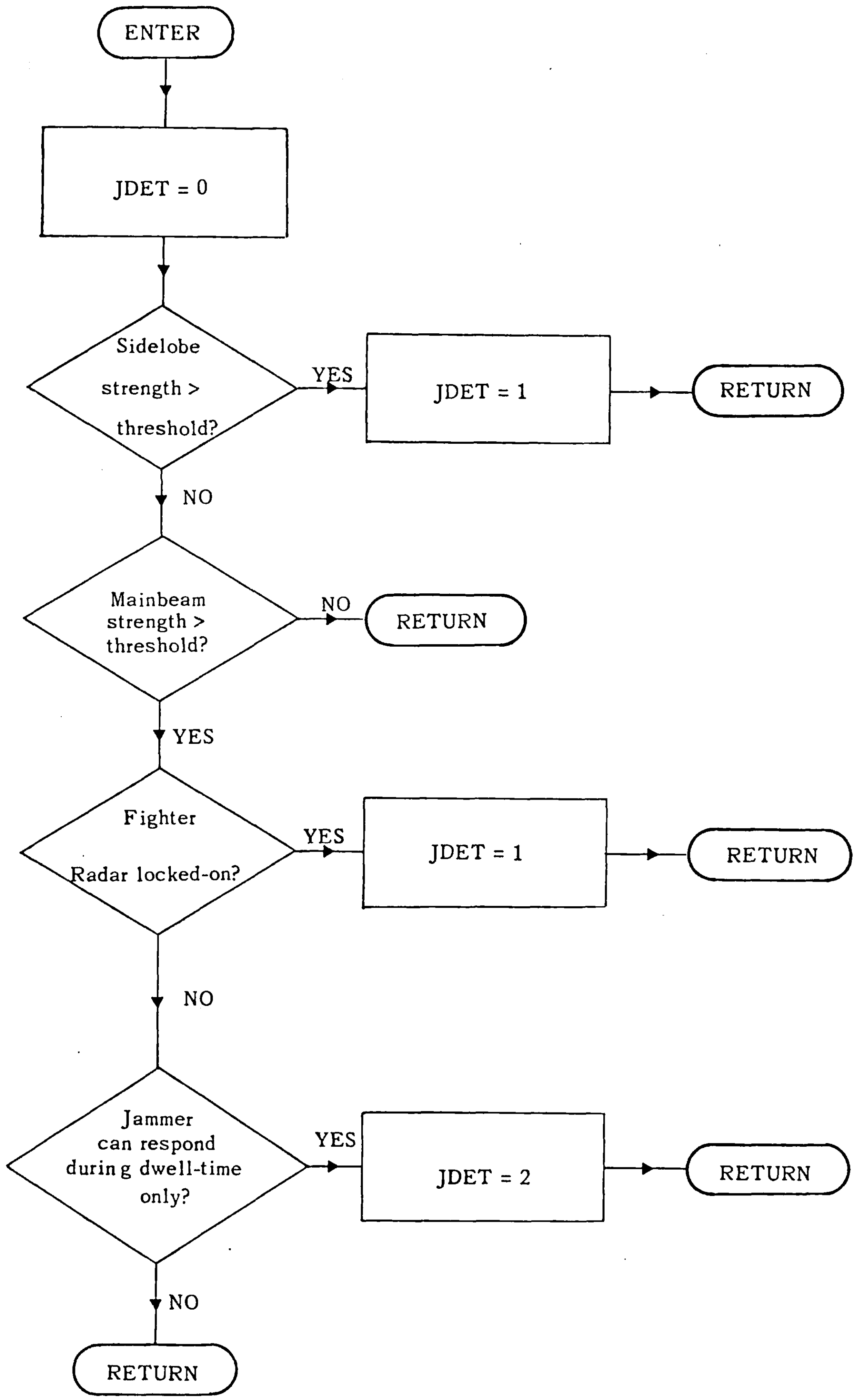


FIGURE 36.1. PROGRAM LOGIC – SUBROUTINE JDETLOGIC

JDET=2 corresponds to the jammer detecting and responding to the fighter's AI radar while it is still in surveillance mode, ie not locked-on. The radar is assumed to be jammed during its paint only.

The program logic is illustrated in Figure 36.1. A detection threshold PDETJ must be specified in the input data for each jammer type. If the sidelobe strength is greater than this threshold, the radar is continuously detected and therefore continuously jammed. The sidelobe strength is specified in the input data, for each AI radar type, simply as a power (in dB) down on the mainbeam strength. If the mainbeam strength (which is calculated in the first section of subroutine RADAR and denoted by POWER) is beneath this threshold, the radar is undetected and, unless it picks up jamming directed at another fighter, it is unjammed by this jammer. If the mainbeam only is greater than this threshold, and the radar is in a locked-on mode the radar will be continuously detected and therefore continuously jammed. (The value of the control variable MODER(IF) indicates whether or not the fighter is in a locked-on mode - see Figure 0.5. MODER=3,10 or 7 if the fighter has fired a missile in clear conditions, Home-on-Jam mode or Angle-on-Jam mode respectively. It is assumed that a fighter is always either locked-on or attempting to lock-on when it fires a missile. Also, if the fighter does not have Track While Scan (TWS) radar, it is assumed that its radar must lock-on to acquire range information (MODER=2), and that it continues attempting to lock-on if forced into a reversionary HOJ mode (MODER=4)). If this is not the case then the radar is still in a scanning mode. For each jammer type IJTYPE the input data specifies in the variable JMPX (IJTYPE) whether it can respond to scanning radars during their paint only. If the jammer has this capability, it jams the radar during its paint only; otherwise, this scanning radar remains unjammed by this jammer (unless by accident).

37. SUBROUTINE JPOWER(JDET)

This subroutine is called from subroutine RADAR to determine DPOWER, the effective jamming emitted in band IB by jammer type IJTYPE in raid IJ against fighter IF when its mainbeam is illuminating raid IR. (Stand-off jammers, which are assumed to transmit isotropically, are considered in subroutine SOJPOWER). This is then added to the total jamming power, POWER, suffered by fighter IF when it is illuminating raid IR; this implicitly assumes that the jammers do not interfere with each other. The subroutine is entered with the value of JDET(IF,IJ,IJTYPE); JDET=0 if this fighter is not (intentionally) jammed; JDET=1 if it is jammed continuously, and JDET=2 if it is jammed during its paint only. The bandwidth, BW, of the jamming is first determined. The principle of the (escort or self-screening) jammer response is that a jammer concentrates its power into the minimum possible continuous frequency band. If just one radar is detected within this frequency band IB it is spot-jammed over a fairly narrow bandwidth, SPOTBW (an input quantity). If several radars are continuously detected within this band the model assumes that jamming power is transmitted continuously over a bandwidth just encompassing the highest and lowest detected frequencies. If jammer type IJTYPE in raid IJ detects any scanning radars and it can respond to these during their dwell-time, if necessary it increases the above bandwidth to jam these radars during this dwell-time only. Finally, it is assumed that the jamming bandwidth BW is at least as great as the AI radar bandwidth, RBW. Thus if IFR denotes the carrier frequency of fighter IF and IFRMIN,IFRMAX respectively denote the minimum and maximum continuously detected frequencies in this band by these jammers, then

$$BW=Max\ (SPOTBW,RBW,IFRMAX-IFRMIN,IFR-IFRMIN,IFRMAX-IFR) \tag{37.1}$$

Note that if these jammers do not continuously detect any radars in this band, IFRMIN and IFRMAX are still at their initial values:

$$\begin{aligned} IFRMIN &= 10^6 \text{ MHz} \\ IFRMAX &= 0 \end{aligned} \tag{37.2}$$

The jamming power, P_1, at fighter IF is first calculated:

$$P_1 = PJ.\left(L_R.L_{Pol}.\frac{\lambda^2}{(4\pi)^2}\right).\frac{G_\phi.G}{R_J^2}.\frac{RBW}{BW} \tag{37.3}$$

where

PJ = total jamming power of jammers of type IJTYPE in raid IJ and band IB

L_R = 1-way AI reception loss

L_{Pol} = Jammer polarisation loss

λ = AI radar wavelength

and

$$PJATR = \left(L_R.L_{Pol}.\frac{\lambda^2}{(4\pi)^2}\right)$$ is calculated in subroutine INPUT.

also G = GRMAX(IFTY)= AI radar mainbeam gain

R_J = range from fighter IF to raid IJ

G_ϕ = GJT(INTPHI,IJTYPE) = Jammer transmitter gain at an angle ϕ (see Figure 50.3,subroutine RADAR)

The effective jamming power, DPOWER, from this source at fighter IF when it is looking at raid IR is then determined. If IJ $\neq$ IR the jamming emitted by raid IJ is assumed to be from a point source and:

$$DPOWER = P_1 . \frac{G_\theta}{G} \tag{37.4}$$

Here θ is the angle projected at the fighter by the raids IR and IJ (see Figure 37.1) and is calculated in subroutine DOTPROD. IANGLE, the angle nearest to θ at which polar diagrams are specified, is then obtained from Function IANG. G_θ = GR1(IANGLE,IFTY) is the AI radar gain of fighter IF in the direction θ. If IJ=IR the resolution of the raid by the fighter's radar is taken into consideration. RESFAC denotes the number of independent glimpses of the raid achieved by the fighter's radar during a single scan and is calculated in subroutine RADAR. In this case:

$$DPOWER = \frac{P_1}{RESFAC} \tag{37.5}$$

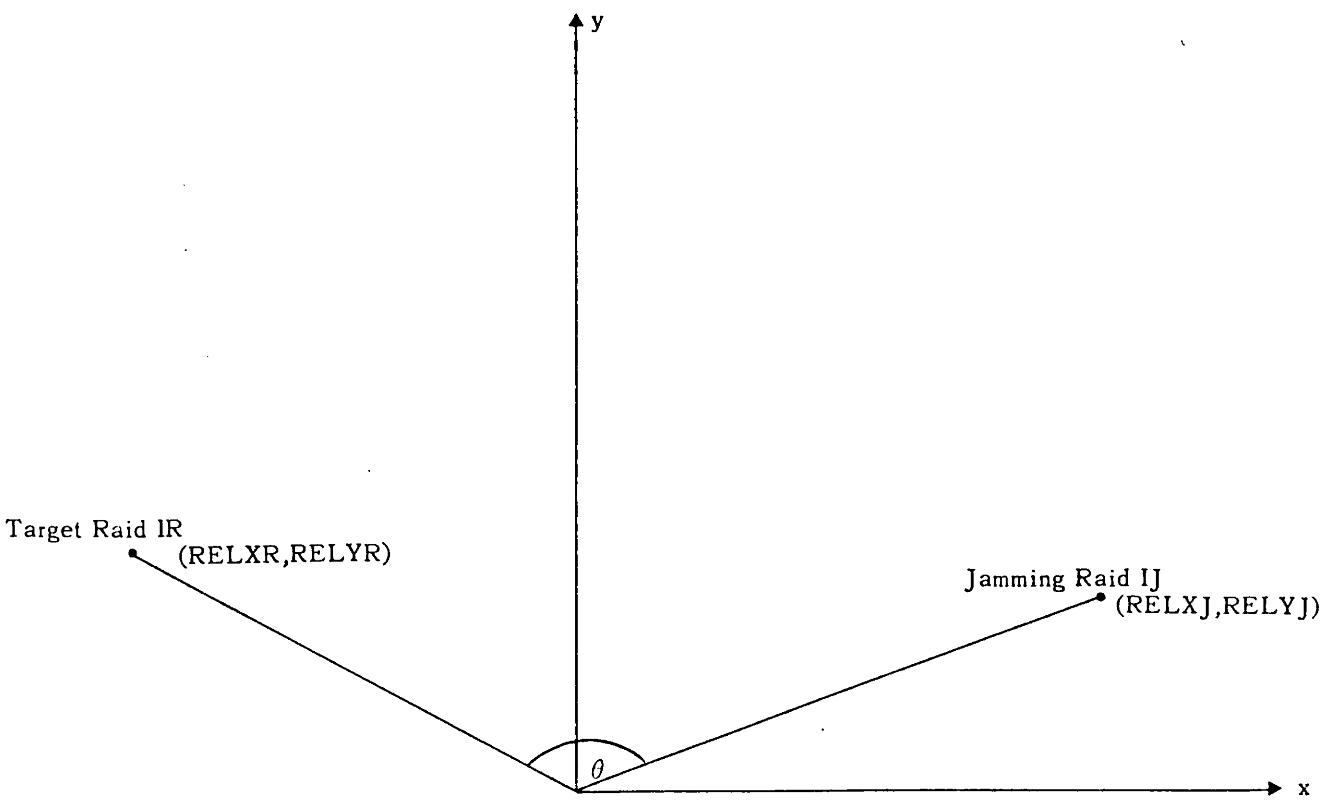


FIGURE 37.1. FIGHTER, RAID AND JAMMER GEOMETRY – SUBROUTINE JPOWER

38. SUBROUTINE LAUNCHTEST

This subroutine is called at each event type 17 to test whether fighter IF is in a position to launch a missile at its target, raid IR. If it is, the necessary processing is carried out and the appropriate splash event, an event type 16, is generated; otherwise another event type 17 is generated a time DTLOOK hence, where DTLOOK is specified in the input data.

The serial IR of the raid being attacked by fighter IF is first recovered from the ITARGET array. Subroutine RELCOORDS then determines the coordinates (RELXR,RELYR) of raid IR relative to fighter IF, and the range RR from the fighter to the raid. Subroutine DOTPROD calculates the angle ϕ illustrated in Figure 38.1, ie the angle between the raid's velocity vector and its position vector relative to the fighter. LAUNCHANGLE, the angle nearest to ϕ at which missile-launch polar diagrams are specified, is then obtained from Function IANG.

The model represents both infra-red and radar missiles, and front and rear hemisphere attacks. Each missile type must have specified a maximum launch-range polar diagram and a corresponding time-of-flight polar diagram, and a minimum launch-range polar diagram. The maximum launch-range may be multiplied by the input variable SAFEFAC to reduce the launch-range (and therefore the corresponding time-of-flight) in order to investigate the effects of 'playing safe'. The missile flight-path is not represented in the model, so that for each of the two missile types the range RR is simply compared with the specified maximum and minimum launch ranges. The current fighter heading is neglected in these calculations.

If neither missile type is within the maximum launch envelope, the next event type 17 for this fighter is generated and control returns to subroutine EVENTS. If just one missile type is within the maximum and minimum launch envelopes it is assumed that a missile of this type is launched and the missile serial is stored in MTYPEF(IF). If both missile types are within the launch envelopes the type with the specified priority is chosen. This depends on the hemisphere of the attack, which is determined by comparing the angle ϕ with the threshold angle ATTACKANGLE, specified in the input data:

If $\phi \leq$ ATTACKANGLE, IHEM(IF)=1,ie forward hemisphere attack

If $\phi >$ ATTACKANGLE, IHEM(IF)=2,ie rear hemisphere attack

Finally, if both missile types are within their minimum launch envelopes the fighter is,currently, assumed to abandon its attack on this raid. Subroutines IATTFIND and IATTDEL remove it from the IATTACK array, ie the list of fighters attacking the raid, and subroutine EV9 provisionally sends it to an appropriate CAP.

If a missile is launched at this entry to subroutine LAUNCHTEST, the time-of-flight is calculated and the missile-splash event, an event type 16, is generated. The number of missiles remaining of the type launched is reduced by one. The missile lethality, P_k, for each missile type is at present a function of two factors:

(i) Whether the attack is forward hemisphere or rear hemisphere (see above)

(ii) The ECM conditions prevailing at the time of launch for infra-red missiles and at missile splash for radar missiles. (The difference is because radar missiles need to be guided during their flight). Three such conditions are represented in the model - clear, Home-on-Jam (HOJ) and Angle-on-Jam (AOJ). The conditions prevailing for fighter IF are determined by the current value of the control variable MODER(IF). The mode of missile type MTYPE is defined by the variable MISSMODE(MTYPE), where MISSMODE(MTYPE)=1 or 2 corresponds respectively to radar or infra-red missiles. If an infra-red missile is launched the variable IRLETH(IF) is set to 0,1 or 2, depending on the current ECM conditions - AOJ,HOJ or clear, respectively. If a radar missile is launched the control variable itself is changed; MODER(IF) is set equal to 7,10 or 3 if current ECM conditions are AOJ,HOJ or clear respectively (see Figure 0.5).

Finally, note that the P_k of a missile will also depend on the altitude difference between the fighter and its target, ie on whether the attack is essentially in level flight, snap-up or snap-down. Until fighter altitude is represented in more detail in the model, this distinction cannot properly be made.

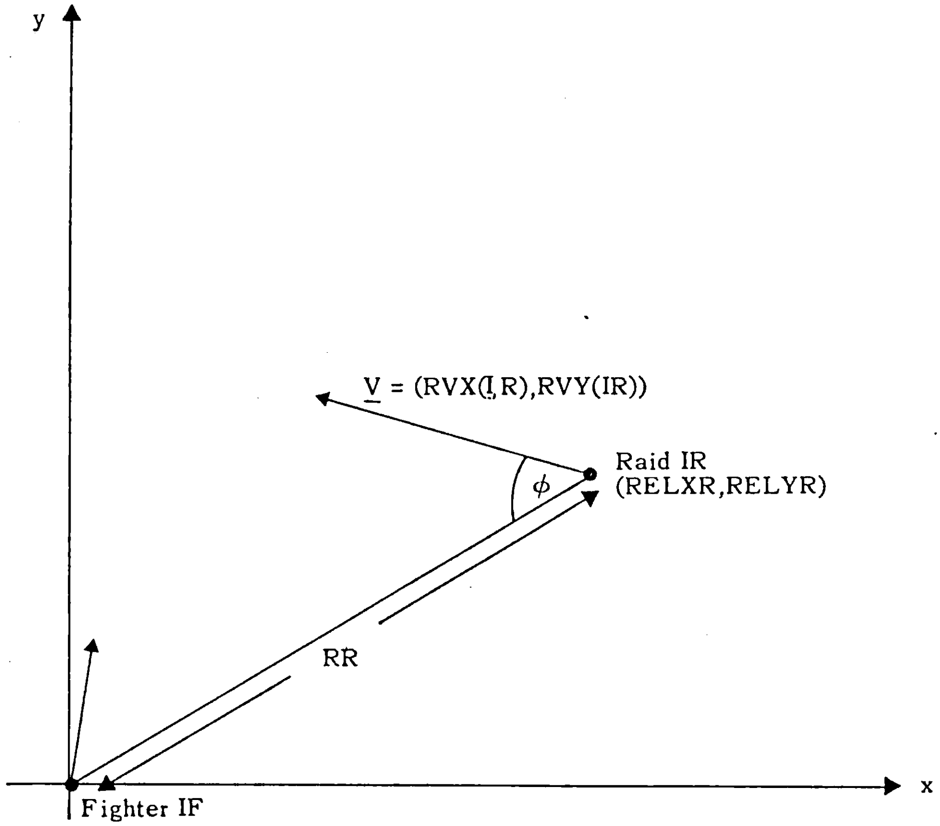


FIGURE 38.1. FIGHTER AND RAID GEOMETRY - SUBROUTINE LAUNCHTEST

39. SUBROUTINE LEADPURSUIT(I)

This routine is called from subroutines RADEVENTS (with I=0) to initiate a lead-pursuit (LP) course for fighter IF against raid IR, and MOVE1F (with I=1) to update this lead-pursuit course.

The fighter and raid geometry is as shown in Figure 39.1. Subroutine RELCOORDS finds the coordinates (RELX,RELY) of the raid relative to the fighter and the range RR from the fighter to the raid. If the LP course is just being initiated (I=0) the fighter's current LP angle, ANGLP(IF), is set equal to the initial LP angle, ALPORIG, specified in the input data. Otherwise, subroutine DOTPROD is called to find the angle ϕ between the raid's position vector relative to the fighter and its velocity vector, as illustrated in Figures 39.1 and 39.2 ($0 \leq \phi \leq \pi$). If $\underline{r}$=(-RELX,-RELY) and $\underline{v}$= (RVX,RVY), ϕ is derived from:

$$\underline{r} \cdot \underline{v} = |\underline{r}||\underline{v}| \cos\phi \qquad (39.1)$$

Note that this expression does not discriminate between positive and negative values of ϕ , as illustrated in Figures 39.1 and 39.2

It is assumed in the following processing that the fighter has some means of measuring the angle ϕ . The variable THILP(IF) is set equal to the current value of ϕ and this is compared with the value of ϕ on the last occasion when the fighter's LP course was updated. If ϕ is increasing the LP course is lagging so that the LP angle α =ANGLP(IF) ought to be increased; similarly, if it is decreasing the LP course is advancing and α should be decreased. Currently, if the increase or decrease in ϕ is greater than 2^o then α is respectively increased or decreased by 2^o.

Having found the fighter's LP angle its course may now be calculated. In order to discriminate between the situations illustrated in Figures 39.1 and 39.2 the dot product $\underline{r}.\underline{v}$ is replaced by the cross-product $\underline{r} \times \underline{v}$. This gives:

$$\text{(-RELX.RVY)-(-RELY.RVX)} = |\underline{r}||\underline{v}| \sin\phi \qquad (39.2)$$

If the L.H.S. of (39.2) is positive this corresponds to Figure 39.2 and the fighter's course is given by a clockwise rotation through an angle α from the raid's position vector. Conversely, if it is negative, this corresponds to Figure 39.1 and the fighter's course is given by an anticlockwise rotation through an angle α from the raid's position vector.

If β denotes the bearing of the raid from the fighter and θ denotes the fighter heading (measured from the positive x-axis), the situations illustrated in Figures 39.1 and 39.2 give, respectively,

$$\theta = \beta + \alpha \qquad (39.3)$$

and

$$\theta = \beta - \alpha$$

The fighter's velocity and acceleration components, (FVX,FVY) and (FFX,FFY) may finally be obtained, knowing the current raid speed FV, and acceleration, FF:

$$FVX=FV.\cos\theta \quad FVY=FV.\sin\theta \qquad (39.4)$$

$$FFX=FF.\cos\theta \quad FFY=FF.\sin\theta \qquad (39.5)$$

40. SUBROUTINE MEANTIMES

This is an elementary routine called from subroutine OUTPUT to calculate the mean values of random variables which are specified in the parameter list.

41. SUBROUTINE MISSREAD

This routine is called from subroutine INPUT and reads in all the missile data, the definitions of which are given in the Glossary. Subroutine GAINREAD is called to read the missile launch envelopes.

42. SUBROUTINE MOVEF

This routine is called from subroutine EVENTS at each radar-scan event (event type 6) to update the position, velocity and fuel state of every fighter.

Subroutine SETIF is first called with parameter list (1,4). This subroutine defines the range of fighters under consideration, and in this case all allocated fighters are considered. Subroutine MOVE1F is then called in turn for each fighter to update its position, velocity and fuel state.

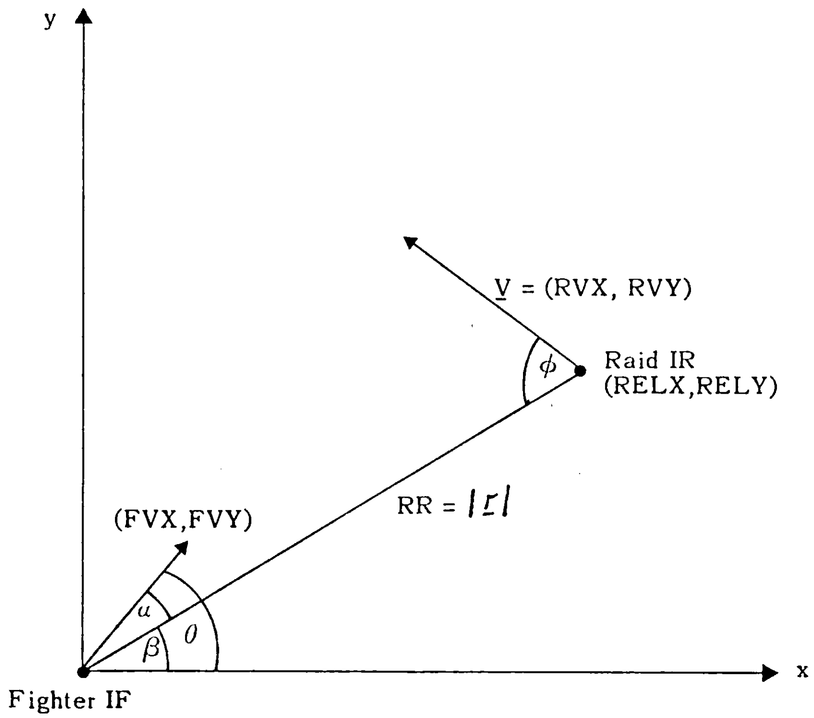


FIGURE 39.1. SUBROUTINE LEADPURSUIT

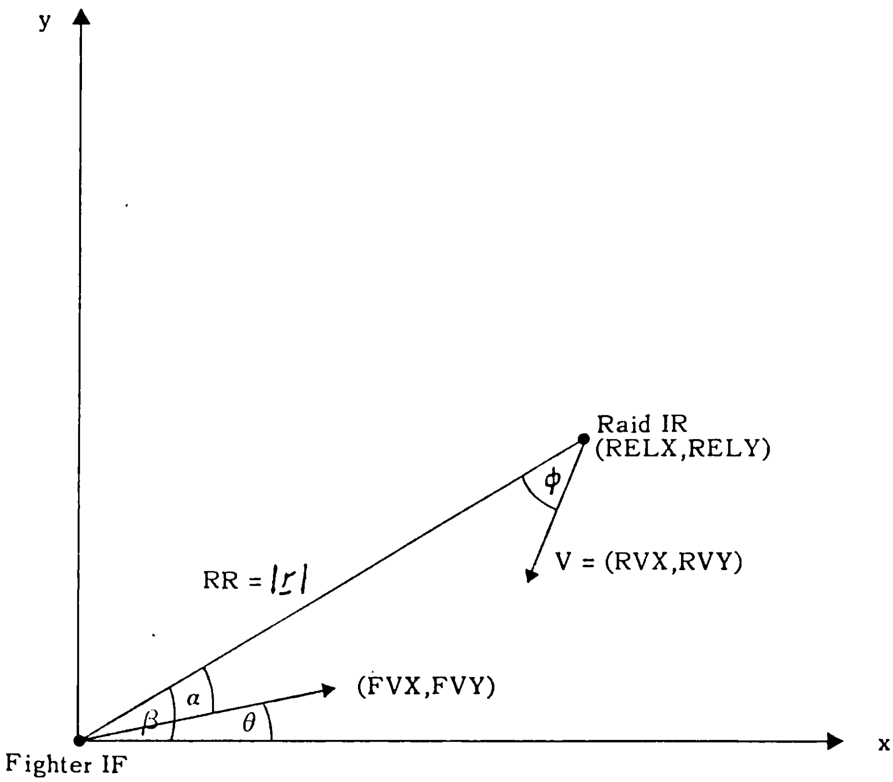


FIGURE 39.2. SUBROUTINE LEADPURSUIT

43. SUBROUTINE MOVE1F

This subroutine updates the position, velocity and fuel state of fighter IF.

If the fighter has not yet taken-off (MODER(IF)=0) control returns immediately to the calling subroutine. If the fighter is turning following an attack, subroutine FIGHTERTURN is called after which control again returns to the calling routine. Otherwise, the time DT since the fighter's track was last updated is calculated:

$$DT=TIME-FT(TF) \qquad (43.1)$$

If f denotes the fighter acceleration, its new speed V_{new} is given by

$$V_{new}=V_{old}+f.DT \qquad (43.2)$$

Note that f is assumed constant, for subroutine MOVE1F is called at each event type 8, ie at each point at which the fighter's acceleration may change.

Similarly, if (VX,VY) and (f_x,f_y) are the components of the fighter velocity and acceleration respectively, then the fighter's x- and y- coordinates are given by:

$$x_{new}= x_{old}+DT(VX_{old}+\tfrac{1}{2}f_x.DT) \qquad (43.3)$$

$$y_{new}=y_{old}+DT(VY_{old}+\tfrac{1}{2}f_y.DT)$$

also

$$VX_{new}=VX_{old}+f_x.DT \qquad (43.4)$$

$$VY_{new}=VY_{old}+f_y.DT$$

The updated fighter position and velocity is then printed, unless this information is suppressed (TIME<TPRINT or IDUMF=0). Finally, if the fighter is following a lead-pursuit course (MODER(IF)=6 or MODER(IF)= 7) subroutine OUTOFB checks that the fighter has not flown into one of the out-of-bounds areas specified in the input data, while subroutine LEADPURSUIT recalculates the fighter's course.

44. SUBROUTINE MOVER

This subroutine updates the position of each raid. Each raid IR is considered in turn; if LOCEVR(IR)=0 the raid has either already been totally destroyed or has reached the end of its defined flight path and is not considered further. Otherwise the time DT since its position was last updated is calculated:

$$DT=TIME-RT(IR) \qquad (44.1)$$

The variable RT(IR) is updated (RT(IR)=TIME) and the raid's x- and y-coordinates calculated:

$$x_{new}=x_{old}+V_x.DT \qquad (44.2)$$

$$y_{new}=y_{old}+V_y.DT$$

where (V_x,V_y) are the current components of the fighter velocity. These may be assumed constant, for the raid position is also updated at each of its track-change points (event type 2). Raids are assumed to travel with constant speed (and height) between track-change points.

Finally, the updated raid positions are printed, unless this information is suppressed. ($TIME<TPRINT$, IDUMR=0, or too short an interval has elapsed since all the raid positions were last printed. Current minimum time between printings = 1 second).

45. SUBROUTINE NEWTRACK

This routine is called from subroutine EVENTS at a raid track change (event type 2) to reset the kinematic track parameters for raid IR, and to generate the next track-change for this raid. If it has reached the end of its defined flight-path, it is deactivated by a call to subroutine RAIDELIM. Any change in track or height at the end of a track-leg is assumed to take place instantaneously. The raid's weapon-release point must be specified as a track-change point, at which no actual change in raid height, speed or track need occur. To avoid unnecessary processing in the calling routine the variable INOCHANGE is set equal to 1 if there is no change in any of the raid parameters, otherwise it is set equal to 0.

46. LOGICAL FUNCTION OUTOFB(X1,Y1,X2,Y2)

This Function tests whether a fighter track starting at the point (X1,Y1) and ending at the point (X2,Y2) takes the fighter into one of the out-of-bounds regions specified in the input data. It is called from subroutine COLLISION when interception courses are calculated and from subroutine MOVE1F to check that lead-pursuit courses do not stray into prohibited areas. The Function simply calls Logical Function INTERSECT to test whether the line from (X1,Y1) to (X2,Y2) intersects any of the specified out-of-bounds lines.

47. SUBROUTINE OUTPUT

This routine is called from the MASTER segment at the end of a run to collect and print statistical information if there is more than one replication. For each random variable on which statistical data is required the following information is derived from the results for each replication: mean value, standard deviation, standard deviation of the mean, minimum and maximum values. Two types of results are produced:

(i) For each fighter IF the statistics described above are calculated for the following events: times of detection, burnthrough and missile-launch; time to missile-splash or collision (depending on whether the fighter's measure of effectiveness is event type 12(collision) or type 16 (missile-splash); this is controlled by the input variable I12OR16); distance of splash or collision from the raid's pre-determined weapon-release point. The basic data from which this information is derived is collected during a model run in subroutine STATS.

(ii) The above results are further aggregated to give the corresponding statistics - times to detection,range and missile-launch; time to collision or missile-splash and corresponding distance of the raid from its weapon-release point - taken over all replications and over all fighters. Again, the mean, standard deviation, minimum and maximum values of these random variables are calculated.

This is essentially a 'stop-gap' routine, for it is envisaged that detailed output routines will be written as required during the course of studies. The output routine as written is based on the fighter serials, which has two major implications:

(i) If fighter IF runs out of fuel or missiles during an interception it is eliminated from the list of active fighters and its serial currently becomes available for reallocation. Hence, if in the model scenario raids attack over a long period of time - longer than the fighter sortie length - the serial IF may refer to more than one fighter.

(ii) If there is a wide variation between replications due to the stochastic processes being modelled, the serial IF need not refer to the same fighter (ie the same fighter type from the same base) on different replications. This problem is structurally more fundamental than that of (i). To eliminate it, statistics would have to be collected not on the basis of fighter serials but in terms of the corresponding invariants - the number of fighters of each type initially available at each base.

48. FUNCTION PDET

This Function is called from subroutine RADEVENTS when the model is running in stochastic mode. It calculates the probability that fighter IF has detected raid IR in the previous time interval DTR. If the fighter's signal-to-total noise ratio when its mainbeam is illuminating this raid is denoted by SNR, then the detection probability P_{det} is given by:

$$P_{det} = 1 - \left(1 - e^{-G^4/SNR}\right)^{\frac{RESFAC.DTR}{TFRAME}} \qquad (48.1)$$

where

G is Swerling's G-factor (input data)

RESFAC is the effective number of azimuth resolution cells enclosing the raid, and is a measure of the number of independent glimpses of the raid achieved per scan (see Figure 50.5 in subroutine RADAR)

TFRAME is the time taken for the fighter's AI radar to complete a single scan (input data).

49. SUBROUTINE PUTEVT(EVTIME,IEVTYP,IUNIT,NEW)

This inserts into the event queue a new event of type IEVTYP, due to occur to entity (ie fighter or raid) IUNIT at time EVTIME. Briefly, the next free cell, NEW, is acquired from the free-cells stack and the values of the new event inserted into the corresponding array elements; thus

EVENT(NEW)=EVTIME

IEVENT(2,NEW)=IEVTYP (49.1)

IEVENT(3,NEW)=IUNIT

The new event is then linked into the event queue, after finding the proper place for it. A minor variation in the application of this event structure in the Fighter Model is that the pointer, NEW, to the event data is returned to the calling routine in the parameter list. If the entity IUNIT refers to a fighter rather than a raid (determined by the event type IEVTYP), this parameter is stored in a variable in the calling routine, namely LOCEVF1(IUNIT), LOCEVF2(IUNIT) or LOCEVF3(IUNIT). This event for fighter IUNIT may then be easily cancelled, if required (see subroutines EVRESET, DELETE and DELEVT).

50. SUBROUTINE RADAR

Subroutines RADAR and RADEVENTS are the most fundamental in the Fighter Model. Subroutine RADAR is called from event type 6, the 'radar scan' event, every DTR seconds. It updates the complete AI radar and ECM picture for all fighters and raids and then calls subroutine RADEVENTS to determine the fighters' responses.

A considerable degree of jammer responsiveness by self-screening and escort jammers is allowed for in the model. To represent this, the subroutine is split into two distinct sections, illustrated in Figures 50.1 and 50.2. The first section determines the fighters' signal strengths at the jammers, so that the response of each jammer to each fighter can be calculated. For each fighter AI radar and jammer pair, the jammer detection logic currently assumed is that:

(i) the radar is undetected and unjammed (unless by accident) or

(ii) the radar is continuously detected and continuously jammed (if its frequency is within the range of jamming frequencies) or

(iii) the radar is in scanning mode and is detected and jammed during its paint only (if its frequency is within the range of jamming frequencies).

The second section of the subroutine then determines the jammers' responses to the number and type of the perceived AI radar signals. The principle of the jammer response is that a jammer concentrates its power into the minimum possible continuous frequency band. Thus if several AI radars are continuously detected within the frequency band of one transmitter, the model assumes that jamming power is transmitted continuously over a bandwidth just encompassing the highest and lowest detected frequencies. The total jamming power experienced by each fighter when its mainbeam is illuminating its target is calculated, together with the (pulse doppler) clutter and the target signal return. In the case of a fighter not yet allocated a target this procedure is carried out for each raid, in the hope of achieving a detection. The signal-to-total noise ratio for each fighter illuminating its target (or a potential target) is then calculated and subroutine RADEVENTS determines the fighter's response, based on this data on signal strength and jamming and clutter. powers.

<u>First Section</u>

Figure 50.1 is a flowchart illustrating the logic of the first section of subroutine RADAR, and the following notes refer to details of this flowchart.

(1) IFREQ1(IJ,IJTYPE,IB) and IFREQ2(IJ,IJTYPE,IB) denote respectively the minimum and maximum AI radar frequencies (in MHz) continuously jammed during this radar scan(ie during this call to subroutine RADAR) in jamming band IB, by jammer type IJTYPE in raid IJ. These are initialised at the beginning of each call to subroutine RADAR.

(2) This subroutine is used to define the range of fighters under consideration. In this case <u>all</u> allocated fighters (ie fighters which have been given an identifying serial) are considered.

(3) The variable IR is normally used when referring to specific raids. In this section of the subroutine it is the effectiveness of the raids as sources of jamming which is being examined, so the variable IJ is used to distinguish raids when they are considered as (point) jammers.

(4) A test is made that raid IJ has not been completely destroyed, or reached the end of its pre-planned flight path (LOCEVR(IJ)≠0).

(5) Currently two responsive jammer types are considered, representing bombers/SSJ and specialist escort jammers. SOJ's are not considered here, for they are assumed to emit continuous barrage jamming and their output is not dependent upon the number, type or frequency of incident AI radar signals.

(6) JDET(IF,IJ,IJTYPE) is a measure of the type of jamming emitted by jammers of type IJTYPE in raid IJ against fighter IF. It is initialised here and calculated later, in subroutine JDETLOGIC.

(7) Subroutine TESTFIGHTER determines if there are any special conditions appertaining to fighter IF (eg although it has been allocated, it might not yet have taken off) such that it is not necessary to continue the calculation of jammer responses to its AI radar signals.

(8) The AI carrier frequency (MHz) of fighter IF is denoted by IFREQF(IF) and is set at take-off. IFREQMIN is the minimum jamming frequency considered in the model and IBW the width of each jamming band (determined in subroutine INPUT). The range of jamming frequencies is currently divided into two (equally spaced) such bands. The particular band IB into which fighter IF's carrier frequency falls is given by

IB= 1+ (IFREQF(IF)-IFREQMIN)/IBW (50.1)

Equation (50.1) in IB is calculated in integer variables, ie any fractional part of the answer is truncated to zero. Hence only if IB=1 or 2 does the fighter's carrier frequency lie within the upper and lower limits of the jamming frequencies.

(9) Subrouting RELCOORDS finds the coordinates (RELXJ,RELYJ) of the raid IJ relative to the fighter IF and the range RJ of the jammer from the fighter.

(10) Subroutine DOTPROD calculates the angle ϕ illustrated in Figure 50.3. This is stored in the array element PHI(IF,IJ).

(11) A test is made that the jamming band IB into which the fighter's radar frequency falls equals 1 or 2 (see (8)) and that the range RJ is less than the maximum for AI radar and jammer interactions specified in the input data (RANGEMAX).

(12) For each jammer type IJTYPE the variable POWER is determined. This is the effective power of fighter IF's radar at raid IJ when its mainbeam is illuminating the raid and is given by:

$$\text{POWER}= \left\{ \frac{(P_p \cdot L_T \cdot L_{pol}).G\lambda^2}{(4\pi)^2 \cdot R_J^2} \right\} . G_\phi \qquad (50.2)$$

where

P_p = AI radar peak power

L_T = AI radar transmission loss

L_{pol} = Jammer polarisation loss

G = AI radar mainbeam gain

λ = AI radar wavelength

R_J = range from fighter to raid (=RJ)

and

$$RPATJ = \frac{(P_p . L_T . L_{pol}). G\lambda^2}{(4\pi)^2}$$ is calculated in subroutine INPUT

Finally

G_ϕ = jammer receiver gain in the direction ϕ (assuming that the mainbeam of the jammer is concurrent with its direction of travel). INTPHI, the angle nearest to ϕ at which polar diagrams are specified, is found from Function IANG, so that

INTPHI=IANG(PHI(IF,IJ)) (50.3)

G_ϕ = GJR(INTPHI,IJTYPE) (50.4)

(13) Subroutine JDETLOGIC is called to determine the value of the variable KDET:

KDET=0 if the fighter is not detected by the jammer

KDET=1 if the fighter is detected and jammed continuously

KDET=2 if the fighter is detected and jammed during paint only

(14) If KDET=1 the upper and lower limits of the fighter frequencies continuously jammed in band IB are adjusted, if necessary.

(15) Control passes from this section of subroutine RADAR when the following information has been derived for each fighter IF and each jammer type IJTYPE in raid IJ:

(i) PHI(IF,IJ)

(ii) IFREQ1(IJ,IJTYPE,IB) and IFREQ2(IJ,IJTYPE,IB)

(iii) JDET(IF,IJ,IJTYPE)

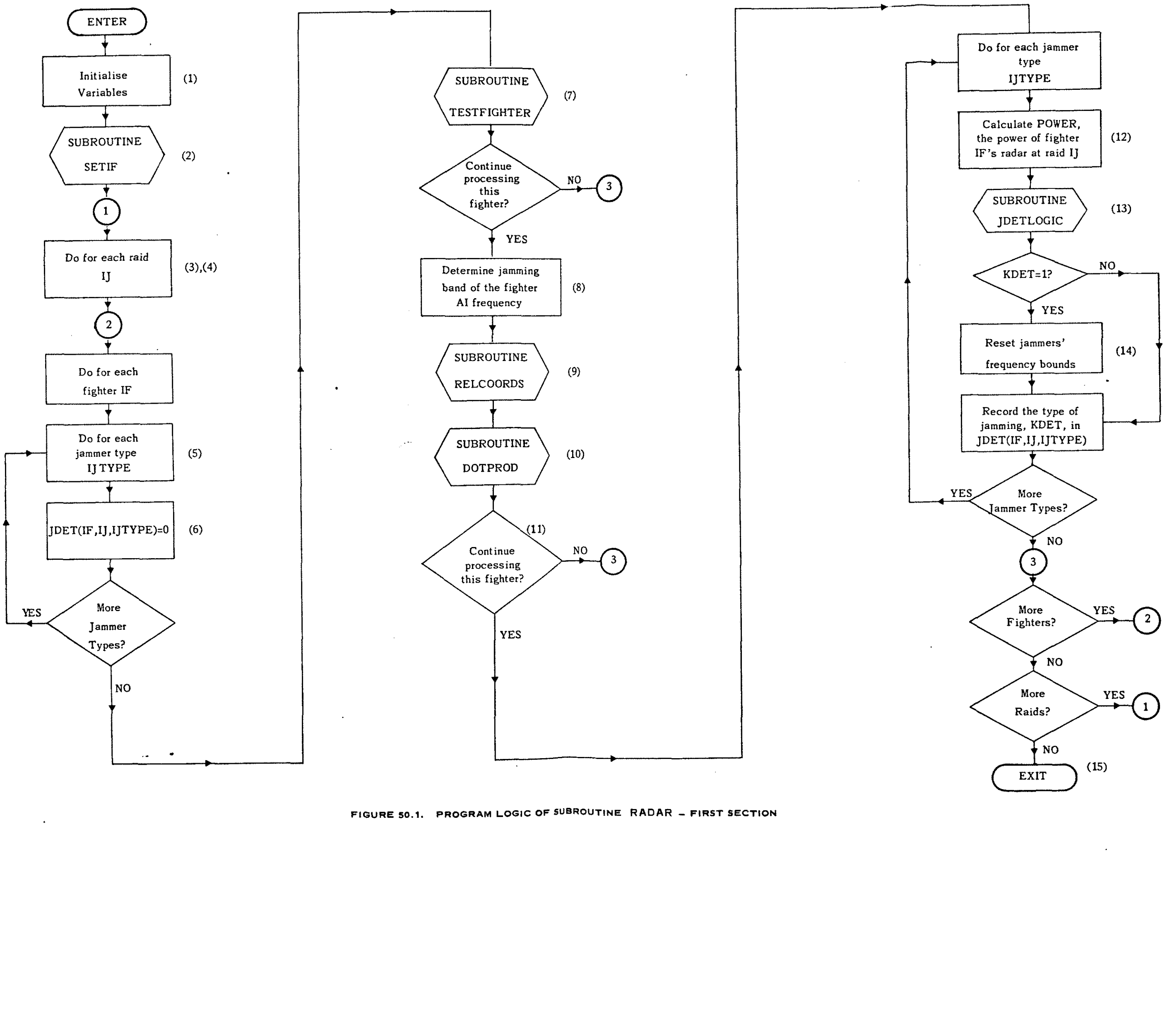


FIGURE 50.1. PROGRAM LOGIC OF SUBROUTINE RADAR – FIRST SECTION

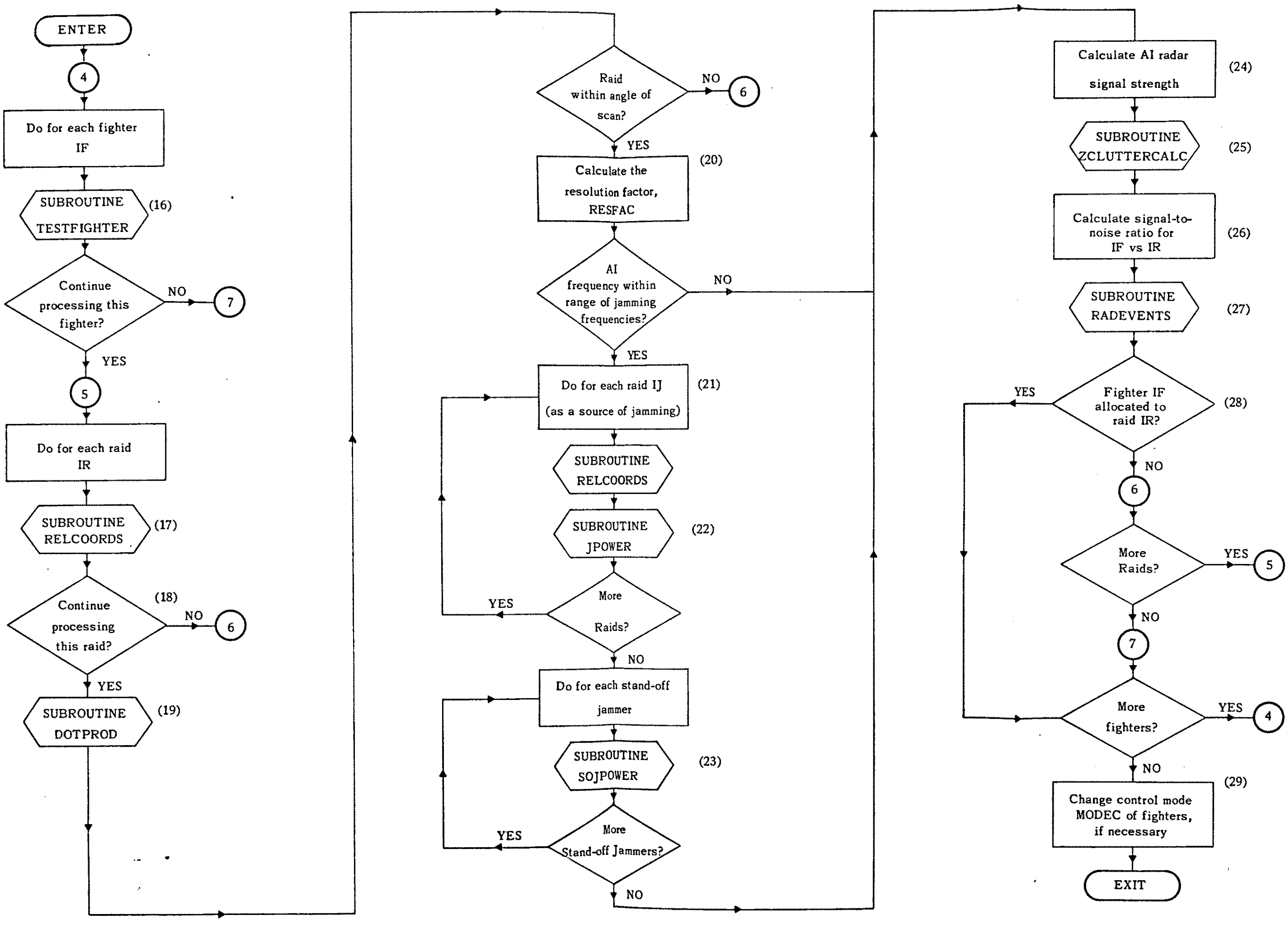


FIGURE 50.2. PROGRAM LOGIC OF SUBROUTINE RADAR – SECOND SECTION

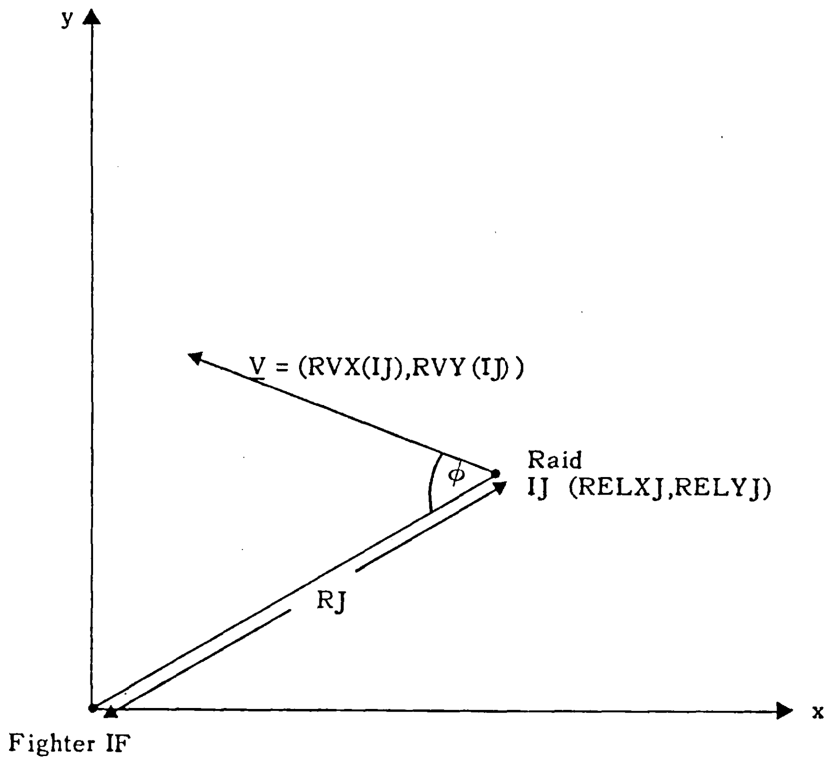


FIGURE 50.3. FIGHTER AND JAMMER GEOMETRY – SUBROUTINE RADAR

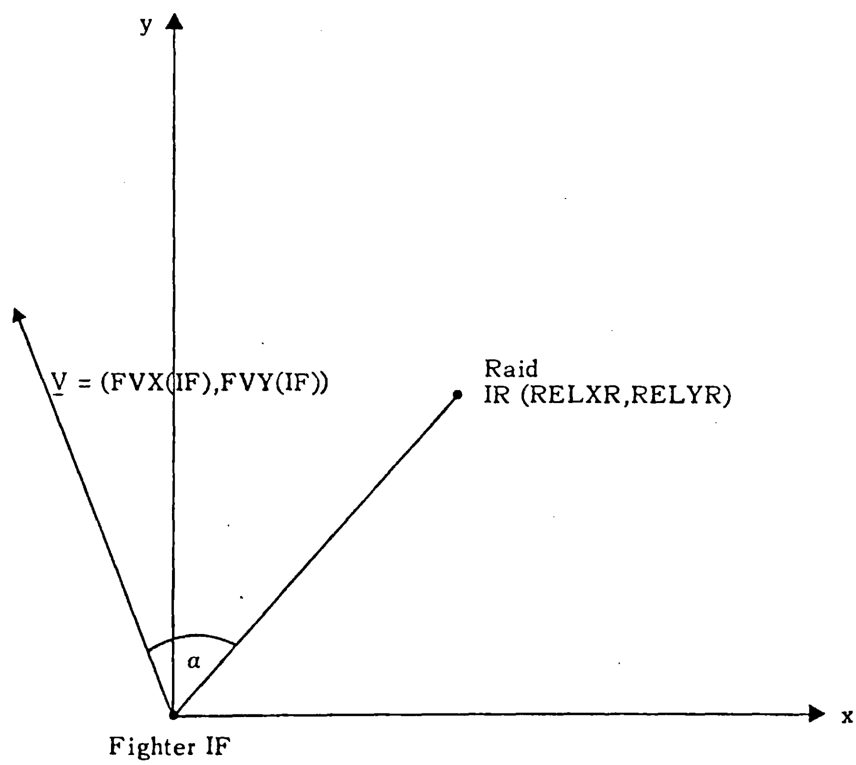


FIGURE 50.4. ANGLE, α, MADE BY THE RAID WITH THE FIGHTER HEADING – SUBROUTINE RADAR

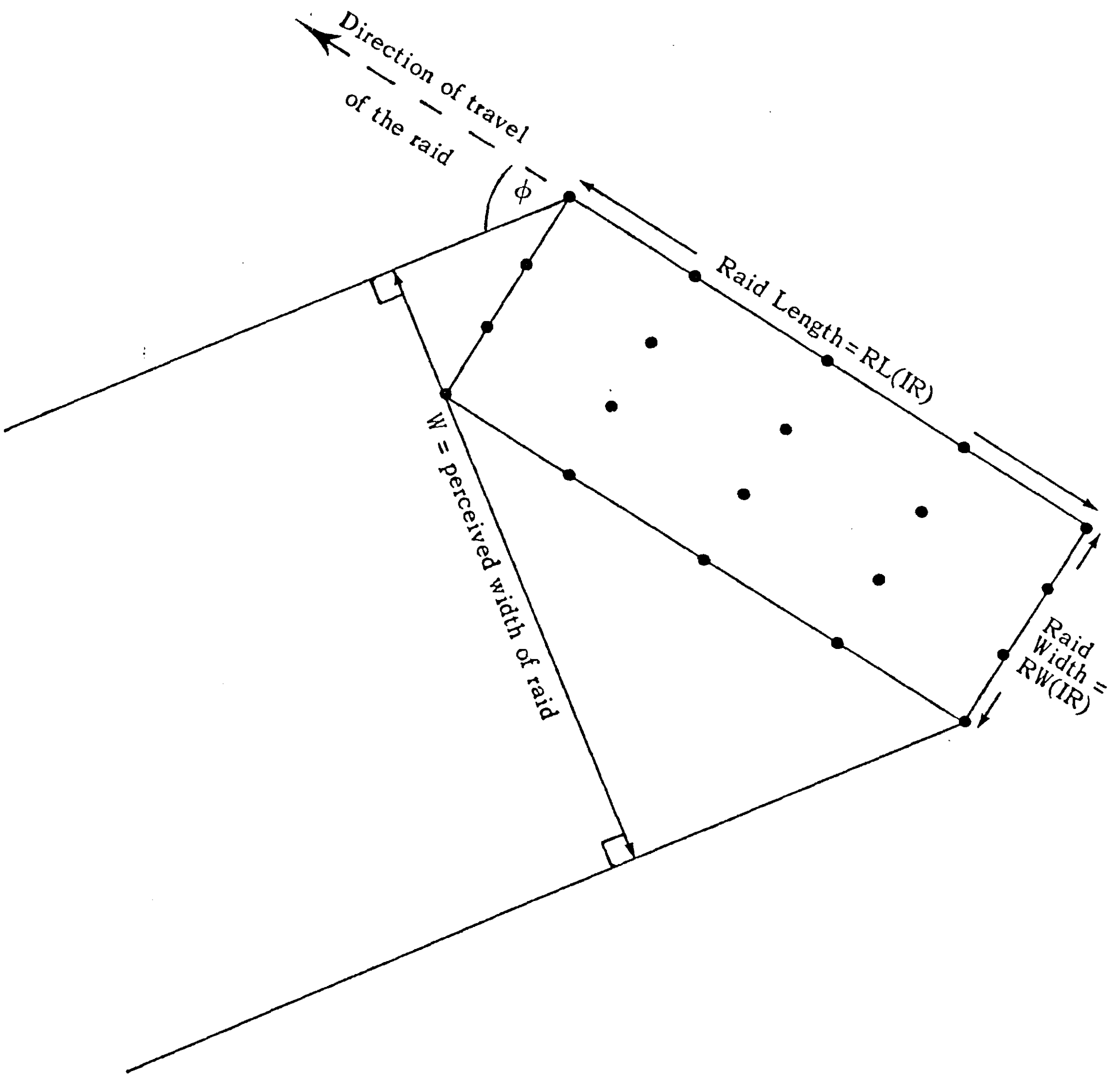


FIGURE 50.5. FIGHTER'S PERCEIVED WIDTH OF RAID IR – SUBROUTINE RADAR

Note: If W_1 denotes the width of the fighter's radar when it illuminates the raid, then the resolution factor RESFAC is defined by:

$$\text{RESFAC} = \max\left\{1, {}^{W}/W_1\right\}.$$

Second Section

Figure 50.2 illustrates the logic of the second section of subroutine RADAR and the following notes refer to details of this flowchart.

(16) Subroutine TESTFIGHTER determines if any special circumstances make further processing of this fighter unnecessary. The fighter's missile state is also checked and the jamming band IB into which the fighter's AI carrier frequency falls is determined.

(17) This determines the coordinates (RELXR,RELYR) of raid IR relative to fighter IF and the range RR from the fighter to the raid.

(18) Further processing of this raid is discontinued if it is not within the maximum range (RANGEMAX) for radar and jammer interactions. There are two additional important conditions under which processing is discontinued:

> (a) If fighter IF is assigned to attack raid IR (so that IR= ITARGET(IF)) and is under Close Control or Data Link Control (MODEC(IF)=3) or is acting autonomously (MODEC(IF)=4), the following processing for fighter IF is conducted only for the target raid IR.
>
> (b) If fighter IF is not allocated to a particular raid (ie it is cruising to or on CAP with MODEC(IF)=1) or is only under Broadcast Control (MODEC(IF)=2) a check is made that the raid currently under consideration as a potential target is not already being attacked by a sufficient number of fighters (NATTACK(IR) is compared with NMAXNO).

(19) Subroutine DOTPROD determines the angle α between the fighter's velocity vector and the position vector of the raid relative to the fighter, as illustrated in Figure 50.4. This is then compared with the AI radar angle of scan, SCANW. If raid IR is outside the angle of scan of fighter IF's radar, normally this fighter and raid pair is not processed further during this call to subroutine RADAR. An exception is made if the fighter has already detected the raid and is attacking it under autonomous control (MODEC(IF)=4). In this case the raid is assumed to have temporarily wandered out of radar cover and the processing of the fighter and raid is allowed to proceed. An appropriate fighter change of course under these circumstances, to bring the raid back within the fighter's radar scan, can easily be modelled if required.

(20) This is the first step in the calculation of the radar signal return, PT, from raid IR to fighter IF. RESFAC is the effective number of azimuth resolution cells enclosing the raid, and is a measure of the number of independent glimpses of the raid achieved by the fighter's radar during a single scan. If W denotes the width of the raid as perceived by the fighter, then from Figure 50.5

$$W=RL(IR).\sin\phi + RW(IR).|\cos\phi| \quad (0 \leq \phi \leq \pi) \tag{50.5}$$

The width of the fighter's radar when it illuminates the raid is W_1, where

W_1=RR. BEAMW (50.6)

and

RR = Range of the raid from the fighter

BEAMW= 3-dB beamwidth of the fighter's radar

The resolution factor, RESFAC, is then given by:

RESFAC=max (1, W/W_1) (50.7)

(21) This section determines the total jamming, denoted by POWER, suffered by fighter IF when its mainbeam is illuminating raid IR (assuming that the jammers do not mutually interfere with each other). Within each raid IJ (considered as a source of jamming) the contribution of each jammer type IJTYPE is assessed separately. In particular, a check is made that jammer type IJTYPE in raid IJ does possess jamming power in band IB, the jamming band corresponding to the fighter's AI frequency (ie PJ(IJ,IJTYPE,IB)>0).

(22). Subroutine JPOWER calculates the power of jammer type IJTYPE in raid IJ at fighter IF when it is looking at raid IR. It makes use of the variable KDET, calculated in the previous section and stored in the JDET array, which describes the type of jamming emitted by jammer type IJTYPE in raid IJ against fighter IF.

(23) Subroutine SOJPOWER calculates the jamming contribution of each stand-off jammer.

(24) The radar cross-section of the raid, as perceived by the fighter, must first be calculated.

If X_1=radar cross-section of bomber/SSJ type (=TXSEC(INTPHI,1))

X_2=radar cross-section of specialist jammer type (=TXSEC(INTPHI,2))

N_1=current number of bomber type aircraft in the raid (=TOTB(IR))

N_2=current number of specialist jammer type aircraft in the raid (=TOTJ(IR))

the fighter's perceived radar cross-section of the raid, denoted by TXSECTION, is given by:

$$\text{TXSECTION} = \frac{N_1.X_1+N_2.X_2}{\text{RESFAC}} \quad (50.8)$$

The AI radar signal return to the fighter from the target (or potential target), raid IR, is then given by:

$$PT = \frac{RPATR.TXSECTION}{(R1)^4} \quad (50.9)$$

Where RPATR is calculated in subroutine INPUT and is given by

$$RPATR = \frac{(P_m.L_T.L_R.L_E).G}{(4\pi)^2} \cdot \frac{G\lambda^2}{4\pi} \quad (50.10)$$

and

P_m = radar mean power

L_T = radar transmission loss

L_R = radar reception loss

L_E = radar eclipsing loss

G = radar mainbeam gain

λ = radar wavelength

(25) This subroutine determines the clutter, PC, experienced by fighter IF when looking at raid IR. Currently it is assumed that the AI radar is operating in pulse doppler (PD) mode, so the PD clutter is calculated. (Pulse clutter routines are also incorporated in the model, though they are not yet used. Subroutine RADAR can easily be modified once a well-defined procedure for the selection of AI radar pulse or PD modes of operation can be specified).

The pulse doppler clutter routine, subroutine ZCLUTTERCALC, calculates the solution PDCL to the integral shown in equation (73.1). The total clutter power, PC is then given by equations (72.15) and (72.17):

PC=CPATR.PDCL/(FV(IF).RH(IR).RH(IR))

where CPATR is calculated in subroutine INPUT and is given by:

$$CPATR = \frac{(P_m.L_T.L_R.L_E).\lambda^3}{4.(4\pi)^3} \cdot \Delta F$$

where

P_m = radar mean power

L_T = radar transmission loss

L_R = radar reception loss

RH(IR)= raid altitude

L_E = radar eclipsing loss

λ = radar wavelength

Δf = radar bandwidth

(26) The signal-to-total noise ratio, SNR, for fighter IF illuminating raid IR, is given by:

SNR=PT/(PC+PN+POWERJ) (50.11)

where

PT = target signal return

PC = clutter

PN = internal radar noise (specified in the input data)

POWERJ = POWER = total jamming power

(27) Subroutine RADEVENTS is a major subroutine which derives fighter IF's response to raid IR, in view of its current state - defined primarily by the control variables MODEC(IF) and MODER(IF) - and the information just calculated on target signal strength and jamming, clutter and noise powers.

(28) Normally, in an entry to subroutine RADAR fighter IF is attacking raid IR, either under Close Control (MODEC(IF)=3) or, after a detection, under autonomous control (MODEC(IF)=4). A fighter only examines each raid as a possible target if it is not already attacking a raid (MODEC(IF)=1 or equivalently MODER(IF)=1) or if it is assigned to a raid but has not achieved a detection and is only under simple Broadcast Control (MODEC(IF)=2 or equivalently MODER(IF)=5). If such a fighter does find a suitable target its control variable MODER is changed immediately, in subroutine RADEVENTS, while the variable MODEC is not altered until the end of subroutine RADAR; hence the test at this point is made on the value of the variable MODER, rather than MODEC.

(29) At the end of this subroutine the control mode MODEC of each fighter is altered if necessary. In the preceding analysis all fighters IF with MODEC(IF)=I are processed before those with MODEC(IF)=J ($1 \leq I < J \leq 4$). Complications would arise, in terms of fighters being processed twice, if MODEC(IF) of fighter IF was increased within the subroutine or within subroutine RADEVENTS. There is only one such increase which can occur, namely from MODEC(IF)=1,2 or 3 to MODEC(IF)=4.

51. SUBROUTINE RADEVENTS

This routine is called from subroutine RADAR for a particular fighter IF and raid IR. It determines the fighter response in view of the current value of the signal return from the raid, PT; the total jamming and clutter experienced by the fighter when its mainbeam is illuminating the raid, POWERJ and PC respectively; the AI radar internal noise, PN and the signal-to-total noise ratio SNR= PT/(PC+POWERJ+PN). All these factors are calculated in subroutine RADAR. Two important parameters used to describe the current state of fighter IF are the control variables MODER(IF) and MODEC(IF). MODER(IF) can take values from 0 to 13 while MODEC(IF) takes values from 1 to 4 - MODEC(IF) is essentially a useful simplification of MODER(IF). The subroutine structure is based on the relationships between the various values of MODER, which are illustrated in Figure 0.5.

The subroutine logic is illustrated in Figures 51.1 and 51.2; the following notes refer to these flowcharts.

(1) This path is followed by a fighter until it achieves a detection.

(2) Thresholds for these tests are specified in the input data. They comprise VISIDENT, the visual identification range; DETSNR, a threshold of the signal-to-total noise ratio for detection in the clear and PJMIN, a threshold of the jammer:noise ratio for detection as a jamming spoke. If the model is in stochastic mode and I12OR16=0 (so that the measure of fighter effectiveness is event type 16 or missile-splash, rather than event type 12 or collision), the detection threshold DETSNR is not used and a probability of detection PDET is calculated instead (in Function PDET). This probability is compared with a random number from a uniform distribution between 0 and 1 to determine if the raid is detected in the clear.

(3) After detection, if I12OR16=0 subroutine PUTEVT sets up the first event type 17 for this fighter; each event type 17 then generates its successor. These events, via subroutine LAUNCHTEST, test the range from the fighter to the raid for possible missile-launch.

(4) It is possible that, if the fighter is under Broadcast Control, the raid which has been detected is not the raid to which the fighter was initially assigned. To change target subroutines IATTFIND and IATTDEL remove IF from the IATTACK array of the original raid (the list of fighters attacking it). IF is inserted into the IATTACK array of raid IR, NATTACK(IR) is increased by one and ITARGET(IF) is set equal to IR.

(5) If the fighter is under CC or Data Link Control and does not acquire range information, it may adopt a lead-pursuit course or continue on its current course (see (14)). In the latter case, to prevent continued testing for detection, MODER(IF) is reset from 11 or 12 to 13.

(6) This corresponds to a fighter cruising to or on CAP and detecting the raid. The fighter changes from profile 1, the cruise profile, to profile 2 the interception profile. It is assumed that the fighter transfers to profile 2 at that point corresponding to its current speed, FV(IF) - see Figure 51.3. JPPT(IF), the serial of the last profile point reached by fighter IF, is updated to correspond to the new profile, and the time when the next change of profile-point (event type 8) is due is calculated in subroutine EV8. The fighter's fuel consumption rate, FRATE, is also updated to take into account the change of profile.

(7) The test for range information is based on a threshold of the signal-to-total noise ratio, SNR. Different such thresholds, TWSSNR and RSNR, may be specified in the input data according to whether the fighter has a TWS radar, or must lock-on to acquire range. If range information is acquired MODER(IF) is set equal to 8 or to 2, depending on whether it is attained via a TWS radar or by lock-on. This distinction is made because of the possibility, in subroutine JDETLOGIC, that a locked-on radar is more likely to generate responsive jamming from the target raid than one which is still in tracking-mode.

Presumably

$$TWSSNR \geqslant RSNR \tag{51.1}$$

if

$$TWSSNR > SNR \geqslant RSNR \tag{51.2}$$

then range information is denied the TWS radar although it would be available if the fighter attempted to lock-on. Nevertheless, it is currently assumed that the fighter does not attempt to lock-on, preferring to wait instead until (hopefully) burnthrough is achieved in TWS mode.

(8) As a result of the fighter's acquisition of range information on the raid, subroutine UPDATEERROR updates the errors applicable to estimates of the raid parameters - position, speed and heading, together with the reaction delay to a raid track-change.

(9) If event type 4 for this raid has not yet occurred, so that GC does not have range and track information, this event is now generated. (In fact an almost identical event type 13 is used, because the information is derived from a fighter rather than from a raid crossing warning line 2 - see subroutine EVENTS).

(10) Subroutine INTCALCONTROL calculates and sets up an interception course for this fighter in subroutine COLLISION and EV12 respectively. Subroutine ERRORS first generates the random numbers used to calculate the errors in the estimates of the raid parameters which are imposed on the interception course calculation.

(11) If the input parameter I12OR16 equals 1 no more radar processing of this fighter is required until it reaches the calculated interception point. The variable ICOLLIDE(IF) is reset from -1 to +1 so that, at each entry to subroutine RADAR, subroutine TESTFIGHTER prevents its continued processing.

(12) If GC has not already resolved the raid size this tests if the fighter is sufficiently close to do so. If it is, the corresponding event (type 5) is generated.

(13) If it is believed that no interception course exists, the fighter abandons its attack on this raid. Subroutines IATTFIND and IATTDEL remove IF from the IATTACK array of fighters attacking raid IR. Subroutine EV9 provisionally sends the fighter to a CAP; in particular, it sets MODEC(IF)=1 and MODER(IF)=1. Hence, on the next entry to subroutine RADAR, every raid will be examined as a potential target for this fighter.

(14) No more processing is carried out if the fighter is on a lead-pursuit (LP) course (MODER=6), or on a CC or DL interception course which, in the absence of range information, it prefers to a LP course (MODER=13 and IGVSF=1). If the fighter was initially cruising to or on CAP (MODER=1), or under BC (MODER=5), or IGVSF=0 (Ground Control course vs Fighter course) then subroutine LEADPURSUIT sets up a lead-pursuit course for this fighter. MODER(IF) is reset to 6 and subroutine DELETE cancels any previously calculated interception event (type 12) or arrival at a CAP point (event type 9 or 10).

15. Tests on missile launch-range are carried out in subroutine LAUNCHTEST so that, from Figure 0.5, the possible changes of fighter state to be considered are as follows. The tests applied when MODER(IF)=2,3,4,7 or 10 on entry to subroutine RADEVENTS are based on RAE (1974a and 1974b)

(i) MODER=2 on entry. Jamming may have increased sufficiently since the last radar-scan event (type 6) to deny the fighter further range information, so that it must revert to Home-on-Jam (HOJ) mode. It is assumed to continue on its interception course, but there is possibly a degradation in missile effectiveness if it is launched in HOJ mode rather than in clear conditions. This is represented by setting MODER=4. Two tests are carried out to determine the fighter state: if the Jamming: Noise ratio, POWERJ:PN is less than a specified threshold, HOJSNR, or the Signal:Jamming ratio, PT:POWERJ is greater than a specified threshold, HOJSTARTS, then the jamming is insufficient to force the fighter into the reversionary HOJ mode and MODER remains equal to 2.

(ii) MODER=3 on entry. From Figure 0.5 this corresponds to a fighter, locked-on in clear conditions, with a radar missile in flight. Again, the only permitted change of fighter state is to the reversionary HOJ mode, in this case MODER=10. Exactly the same tests as in (i) above are applied to determine if this change of mode is necessary.

(iii) MODER=4 on entry. This corresponds to a fighter in HOJ mode, attempting to lock-on. The only possible change is to MODER=2 if burnthrough is achieved, either if the Jamming:Noise ratio is less than the specified threshold value HOJEND, or if the Signal: Jamming ratio is greater than the threshold HOJENDS.

(iv) MODER=10 on entry. This corresponds to a fighter with a radar missile in flight in HOJ mode. The only permitted change is to MODER=3 if burnthrough is achieved and the same tests as in (iii) are applied.

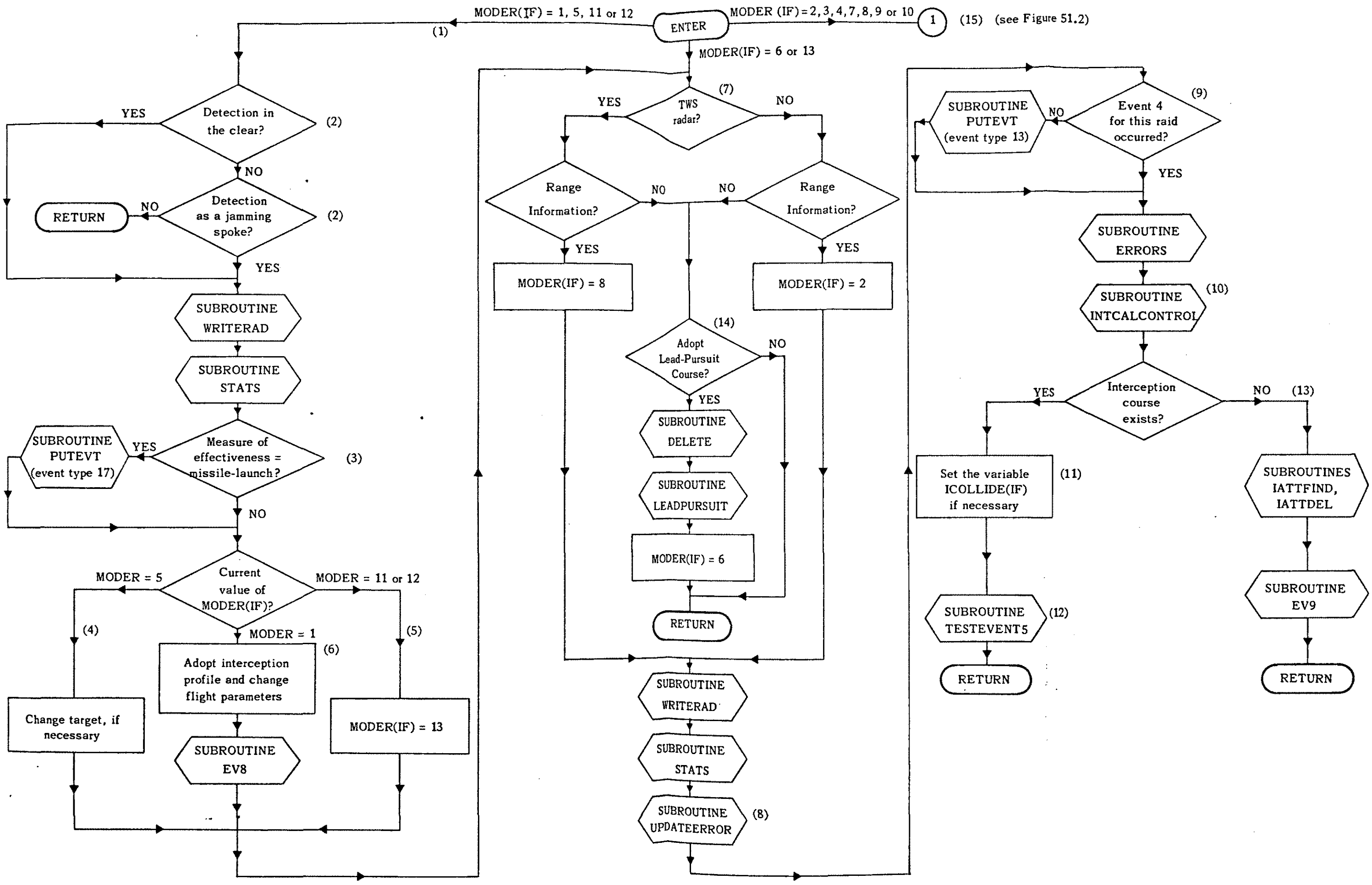
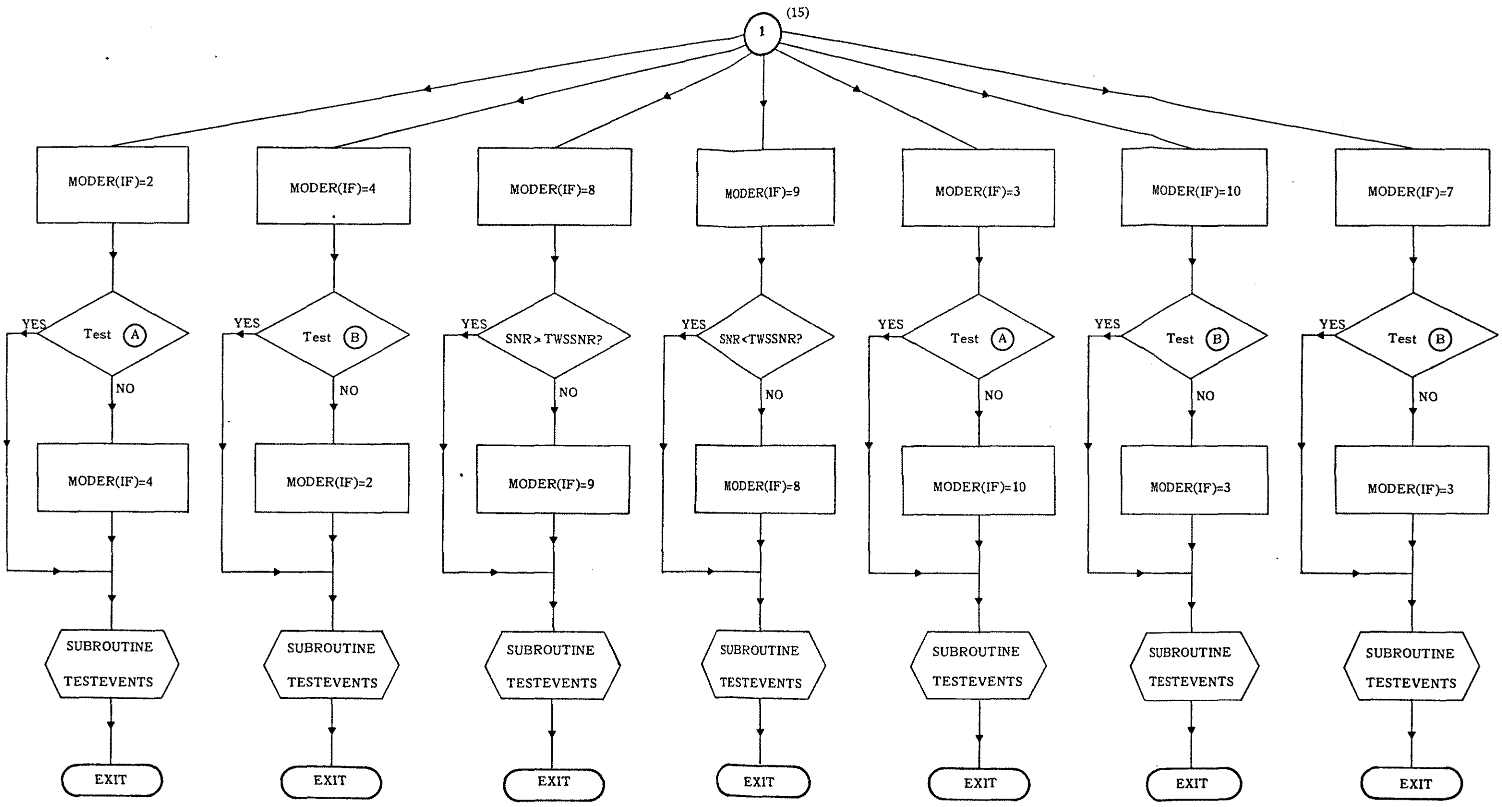


FIGURE 51.1. PROGRAM LOGIC - SUBROUTINE RADEVENTS (PART (i))



Notes
Test (A) : Jamming:Noise ratio,POWERJ:PN, less than HOJSNR <u>or</u> Signal:Jamming ratio, PT:POWERJ, greater than HOJSTARTS
Test (B) : Jamming:Noise ratio,POWERJ:PN, greater than HOJEND <u>and</u> Signal:Jamming ratio, PT:POWERJ, less than HOJENDS

FIGURE 51.2. PROGRAM LOGIC – SUBROUTINE RADEVENTS (PART (ii))

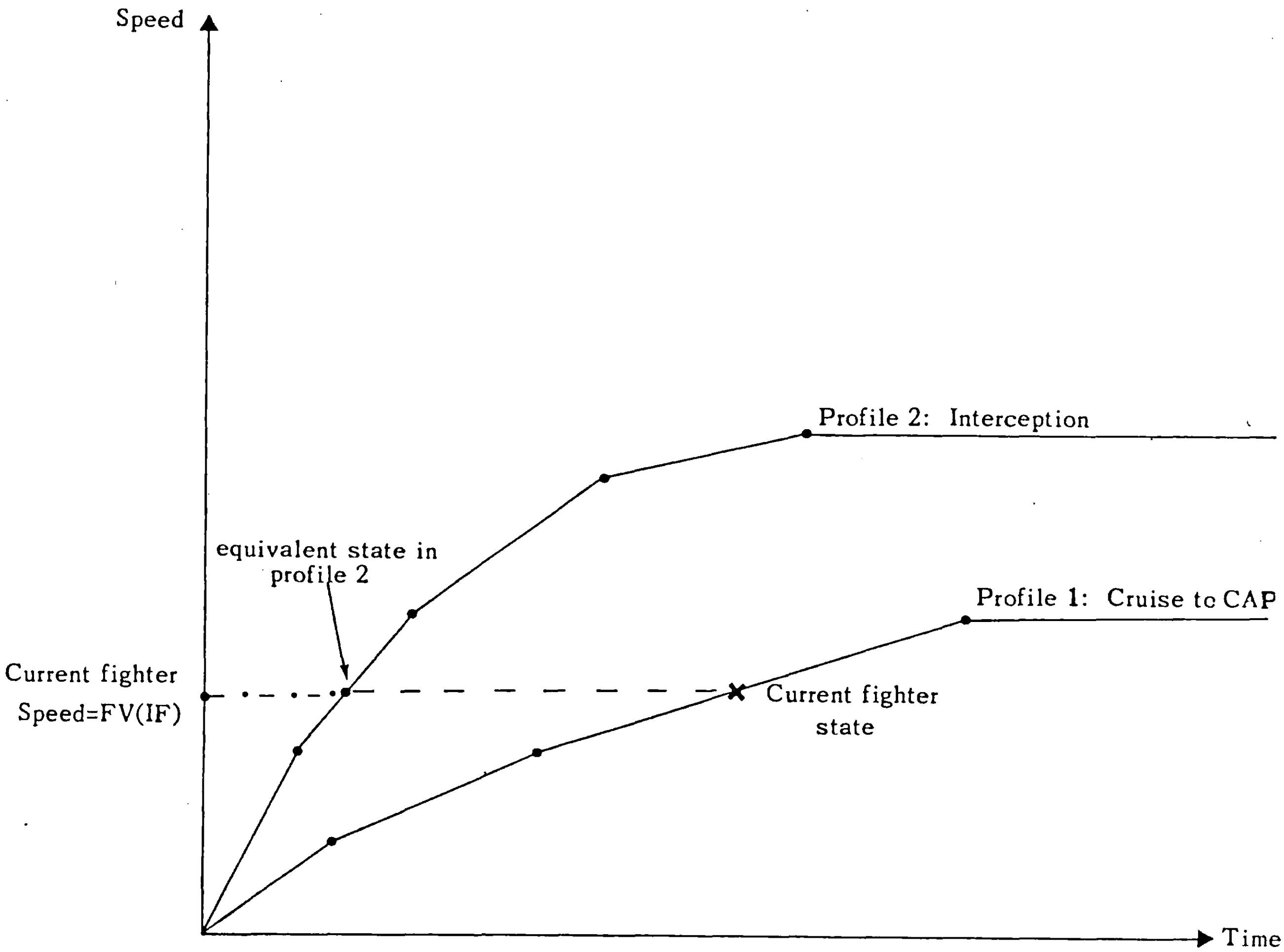


FIGURE 51.3. CHANGE FROM PROFILE 1 TO PROFILE 2 – SUBROUTINE RADEVENTS

(v) MODER=8 on entry if the fighter has acquired range information without locking-on, with its radar in TWS mode. The current signal-to-total noise ratio, SNR, is compared with the threshold TWS signal-to-total noise ratio, TWSSNR, to determine if range is still available to the fighter in TWS mode; if not it reverts to HOJ mode with MODER=9. (The representation of the fire control system is based on the Phantom and Tornado; it is not yet clear whether this test or the test in (i) is most appropriate here).

(vi) MODER=9 on entry if the fighter is in a reversionary HOJ mode with its radar in TWS mode, so that it has not locked-on. The current signal-to-total noise ratio, SNR, is compared with the track-while scan threshold value, TWSSNR, to determine if the fighter has achieved burnthough since the last radar scan; if so MODER is reset to 8. Again, it is not clear whether this test or that of (iii) is most appropriate.

(vii) MODER=7 on entry if the fighter has a radar missile in flight in AOJ mode. The only possible change of mode is to MODER=3, if the fighter achieves burnthrough. It is currently assumed that the burnthrough criteria are the same for AOJ as for HOJ, so the same tests as in (iii) and (iv) are applied.

52. SUBROUTINE RAIDELIM

This routine is called from subroutines NEWTRACK and RAIDKILL to eliminate raid IR, either because it has reached the end of its defined flight-path or because it has been totally destroyed. Any fighters attacking the raid are removed from the IATTACK array (ie the list of fighters attacking the raid) in subroutine IATTDEL and provisionally sent to CAP in subroutine EV9. While flying to CAP the fighters' radars continue scanning, so that they have the opportunity to detect and attack further raids.

53. SUBROUTINE RAIDKILL(MPK1)

This routine is called from subroutine EVENTS at an event type 16, ie a missile splash by fighter IF against raid IR, to reduce the number of aircraft remaining in the raid. In the stochastic mode of operation a random number, weighted by the current relative densities of bombers and jammers, is used to decide whether a bomber or a jammer has been killed. (In the stochastic mode the test to determine whether a kill has occurred is carried out in the event 16 processing; if there is no kill, the call of subroutine RAIDKILL is omitted). If MPK1 denotes the missile kill probability (specified in the parameter list), in the deterministic mode a kill of size MPK1 always occurs, so that flight and raid sizes and numbers of bombers and jammers may be fractional quantities. Again, the proportions of bombers and jammers killed depend on their current relative densities. Finally, with the elimination of (fractional or integer) bombers and jammers from the raid, the total jamming power of the raid must be reduced accordingly.

For most purposes the fighter model considers a raid to be a collection of enemy aircraft collected at a single point. However, when assessing the damage caused by a missile splash and choosing subsequent targets for the fighter in the REATTACK routine, greater detail is required. In these cases the raid is considered as a string of flights, each flight being a known mixture of bombers and jammers with its centre of gravity a known distance from the raid's centre of mass (see Figure 0.1) When a fighter makes a forward attack on a raid and a kill occurs, it is taken that an aircraft from the leading flight is killed. Similarly a rear attack affects the trailing flight.

The subroutine logic is illustrated in Figure 53.1 and the following notes refer to details of this flowchart.

(1)(a) When a fighter makes its initial attack on a raid no specific flight has been assigned as its target, so that IFLIGHT(IF)=-1. The flight to attack is chosen as follows:

(i) Front hemisphere attack - the first non-empty flight from the front of the raid

(ii) Rear hemisphere attack - the first non-empty flight from the rear of the raid.

Note that a flight's serial number remains the same as that defined on input, regardless of whether or not other flights have been destroyed.

(b) If the fighter is making a second or subsequent attack on the raid it will already have been assigned a flight as its target.

(2) In the stochastic mode the type of enemy aircraft killed depends upon their current relative densities. If there are N_B bombers and N_J specialist jammers in the flight being attacked, a random number uniformly distributed between 0 and 1 is found and compared with the quantity $\frac{N_J}{N_B+N_J}$. If the random number is less than this quantity a jammer is assumed killed, otherwise a bomber is assumed killed. Note that this method assumes that bombers and jammers are equally likely to be attacked, regardless of whether the attack is in clear conditions, Home-on-Jam or Angle-on-Jam mode.

(3) If MPK1 denotes the missile kill probability in deterministic mode a kill of size XPK is assumed to occur, where

XPK=min(MPK1, no. of surviving aircraft in the target flight) (53.1)

If there are N_B bombers and N_J jammers in the flight being attacked the bomber and jammer losses are assumed proportional to their relative densities:

$$B_{loss} = \frac{N_B}{N_B+N_J}$$

$$J_{loss} = \frac{N_J}{N_B+N_J} \qquad (53.2)$$

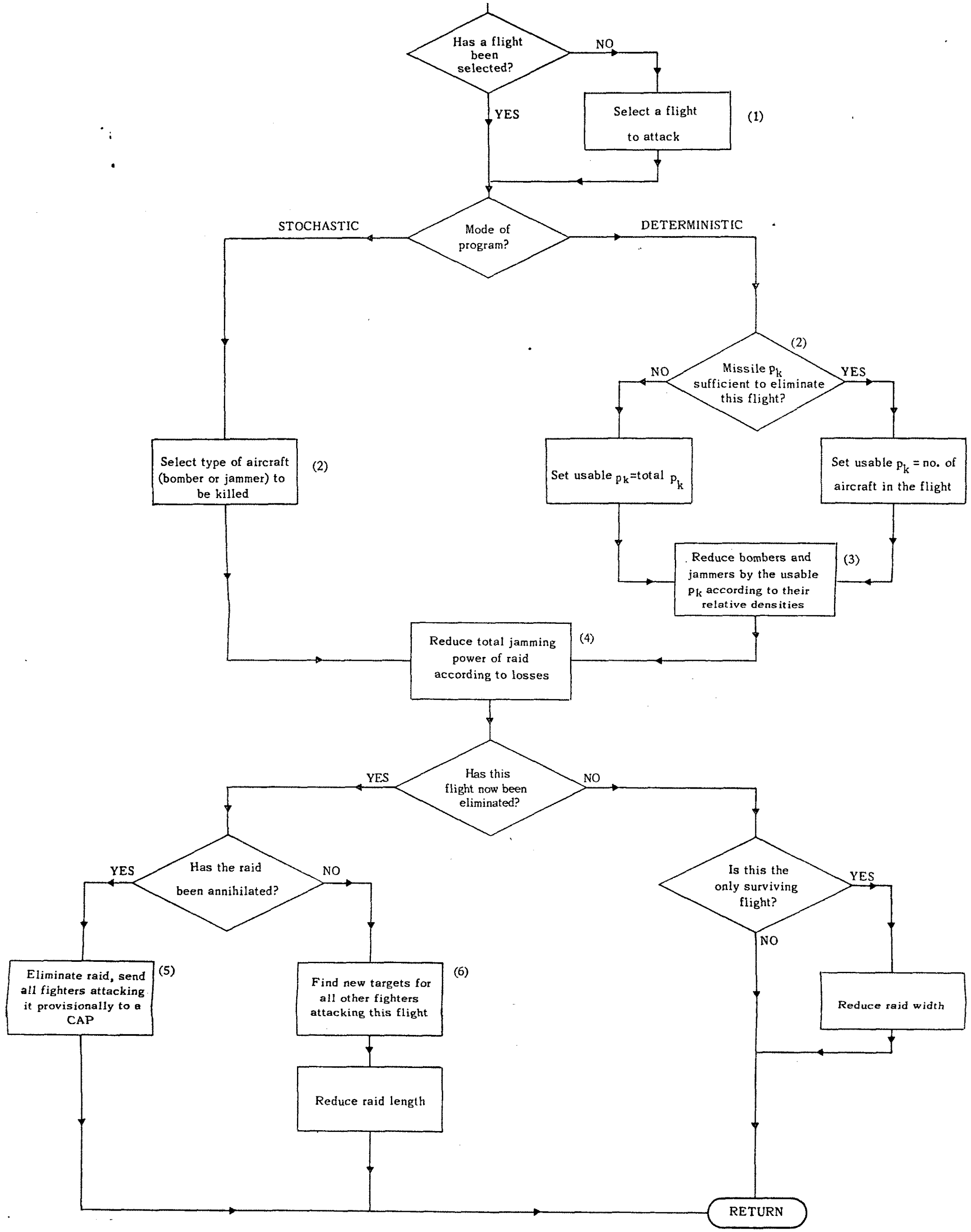


FIGURE 53.1. PROGRAM LOGIC – SUBROUTINE RAIDKILL

This again assumes that bombers and jammers are equally likely targets for the missile.

(4) Although the jamming power of the raid is reduced as a result of aircraft being destroyed, the effect on the spatial distribution within the raid of the remaining jamming is not considered.

(5) If the raid is finally annihilated all fighters attacking it are provisionally sent to CAP in subroutine RAIDELIM. Their AI radars continue scanning, so they may detect and attack further raids via subroutine RADAR.

(6) More than one fighter attacking the raid may have been assigned to this particular flight. If the flight is now completely destroyed these fighters are assigned new flights within the raid to attack in subroutine REATTACK.

54. SUBROUTINE REATTACK

This routine is called from subroutine RAIDKILL and subroutine EVENTS at an event type 16(missile-splash) to set up a reattack for fighter IF on raid IR. In order to measure fighter effectiveness an appreciation is needed of the ability of a fighter to attack and kill enemy aircraft after its initial attack on a raid. The degree of sophistication used in the Fighter Model to represent a reattack sequence is felt to be sufficiently high to be consistent with the rest of the model, yet not so high that its computation and detail dominate the simulation. Thus targets are as usual chosen on the basis of least-time-to-intercept, within fuel limitations. The raid structure is represented in more detail, inasmuch as the individual flights of the raid currently being attacked are assessed as potential targets. Finally, the time taken for the fighter to turn from its current heading onto an interception course against any flight within the raid is taken into account. In the calculation of interception courses in the rest of the Fighter Model (in subroutines EVENTS,FCHOOSE, FFIND or RADEVENTS), each raid is regarded as concentrated at its centre of mass while the time for a fighter to turn onto an interception course is assumed to be negligible compared with the total flight time to interception.

In subroutine REATTACK each flight IFL is considered in turn as a potential target, initially assuming the fighter can turn instantly onto the appropriate attack heading. Subroutine INTCALCONTROL determines (via subroutine COLLISION) whether an interception course exists under this optimistic assumption. (Subroutine ERRORS first generates the random numbers used in subroutine INTCALCONTROL to derive the errors in the estimates of raid position, speed and heading). If so, it calculates the approximate time taken to turn onto this heading. The time to turn, time to intercept (excluding turn time) and flight serial are stored in the I-th position of the arrays TTURNØ,TMINØ,IFLIGHTØ respectively, where the interception times are stored in ascending order:

$$\mathrm{TMIN\emptyset(J)} \leqslant \mathrm{TMIN\emptyset(I)} \leqslant \mathrm{TMIN\emptyset(K)}, \quad 1 \leqslant \mathrm{J} \leqslant \mathrm{I} \leqslant \mathrm{K} \leqslant \mathrm{NFL\emptyset(IR)} \qquad (54.1)$$

(NFLØ(IR) is the number of flights in the raid at the start of each replication; currently NFLØ(IR) ⩽ 7).

If no interception course can be found against any flight within this raid, subroutine REATTACK does not carry out the processing to determine if the fighter can attack any other raid. Instead it is provisionally sent to CAP (in subroutine EV9) and its serial deleted from the list of fighters attacking this raid in subroutines IATTFIND and IATTDEL. Another target for the fighter to attack may then be found at subsequent radar-scan events, in subroutine RADAR.

Otherwise, suppose K flights are determined as potential targets by the above calculations (1 ⩽ K ⩽ NFLØ(IR) ⩽ 7). Subroutine INTCALCONTROL then determines if an interception course exists against the apparent 'best' flight, IFLIGHTØ(1), if the time taken to turn onto the first approximation to an interception course, TTURNØ(1), is taken into account. In the unlikely event that the turn time is sufficiently great as to prevent the fighter making an attack, the next flight in the IFLIGHTØ array is considered. In the extreme, if no suitable flight to attack can be found, the fighter is again provisionally sent to CAP in subroutine EV9. Note that the movement of the fighter in space during its turn is currently neglected (see subroutine FIGHTERTURN), so that the fighter is assumed to spend its calculated turn-time in simply turning 'on the spot'. Subroutine COLLISION determines the angle through which the fighter turns and an approximate turn time is found by assuming a rate 'n' turn, where 'n' is defined on input.

If the fighter can intercept flight IFL when the turn-time is taken into account the interception event (type 12) and change of fighter heading are set up in subroutine EV12 via subroutine INTCALCONTROL. The serial IFL is stored in the variable IFLIGHT(IF) and an event type 18 (end-of-turn event) is generated, to occur at the appropriate time hence. During this turn the fighter's position is frozen, so that at the end of the turn the fighter immediately continues on its calculated interception course and begins testing for a missile-launch opportunity. Finally, the next change of profile-point (event type 8) is delayed by the turn-time, while the control variable MODER(IF) is reset in accordance with Figure 0.5. (The fighter is assumed to rely only on its own information during reattacks so that MODER(IF)=13 is not allowed. If the previous attack was in AOJ mode without range information (MODER(IF)=6,7 or 13) the fighter is assumed to provisionally adopt a lead-pursuit course (MODER(IF)=6) until it achieves burnthrough or again launches in AOJ mode. The planned interception event is cancelled in subroutine DELETE, so that in this case the foregoing analysis may be regarded simply as a means of choosing the next flight to attack).

55. SUBROUTINE RELCOORDS(I,RELX,RELY,RELD)

This subroutine calculates the coordinates (RELX,RELY) of a raid relative to some point and its distance RELD from that point. It is called under three circumstances, dependent upon the value of the control variable, I:

(i) I=0 when called from subroutines INTCALCONTROL and WRITERAD to find the coordinates of raid IR relative to the origin of coordinates.

(ii) I=1 when called from subroutines LAUNCHTEST and RADAR to find the coordinates of raid IR relative to fighter IF and the distance from the fighter to the raid.

(iii) I=2 when called from subroutine RADAR to find the coordinates of raid IJ relative to fighter IF and the distance from the fighter to the raid.

If fighter IF is attacking raid IR or IJ (so that ITARGET(IF)=IR or IJ) in the reattack sequence, it will be allocated to a specific flight (IFLIGHT(IF)). In that case the raid coordinates refer to the centre of mass of the particular flight being attacked, rather than the centre of mass of the raid (see Figure 0.1).

56. SUBROUTINE REPINIT

This routine is called from the MASTER segment at the beginning of each replication to initialise all the variables which may be altered during a model run: current time, size and jamming power of each raid, number of fighters attacking each raid, number of fighters available at each base, etc; all such variables are defined in the Glossary. Subroutine REPINIT also sets up the initial events for each replication:

(i) the end-of-replication (event type 1) at time TEND;

(ii) the start of each raid track, regarded as a track-change point (event type 2) at time TR(IT-1,IR), where

$$TR(IT-1,IR) \leq TSTART < TR(IT,IR) \quad (56.1)$$

The time TR(IT,IR) at which raid IR reaches track-change point IT is calculated in subroutine INPUT via subroutine RTRACKS;

(iii) the events 3,4 and 5 for each raid, corresponding to the times - EWT(IR,IEW), calculated in subroutine INPUT - at which raid IR crosses warning lines IEW=1,2 and 3 respectively;

(iv) finally, the first radar-scan event (type 6) is set up here to occur at time TSTART.

57. SUBROUTINE RESPACC(TDELAY,HEADERR)

This routine serves two distinct purposes:

(i) It may be called from subroutine EVENTS at an event type 2 to determine the delay TDELAY between a track-change of raid IR and the corresponding change of interception course by fighter IF, which is attacking the raid.

(ii) It may be called from subroutine EV12 to determine the heading error HEADERR to be imposed on an interception course for fighter IF.

The fighter heading error and reaction delay to a raid track-change depend on the current control modes MODEC(IF) and MODER(IF) of the fighter. Subroutine RESPACC is only called for fighters either already on, or about to adopt, interception courses, so that MODEC(IF)≠1; the four possible control modes are therefore as follows:

(i) Broadcast Control: ICTYPE=MODEC(IF)=2

(ii) Close Control: ICTYPE=MODEC(IF)=3

(iii) Autonomous control: ICTYPE=MODEC(IF)=4

(iv) Data Link Control: ICTYPE=1; MODER(IF)=12

The heading error is simply set equal to the appropriate element in the HERROR array, which is read in as input data:

HEADERR=HERROR(ICTYPE) (57.1)

The total reaction delay to the track-change of raid IR is the sum of two delays:

(i) The Ground Control processing delay, TRESPONSER(IR)

(ii) The ensuing fighter reaction delay, CDELAY(ICTYPE)

Thus

TDELAY=TRESPONSER(IR)+CDELAY(ICTYPE) (57.2)

Equation (57.2) is modified under two circumstances: if the fighter has not yet taken off (MODER(IF)=0) it is assumed that only the GC processing delay applies, so that TDELAY=TRESPONSER(IR); if the fighter is under autonomous control, having detected the raid (MODEC(IF)=4), it is assumed that only the fighter reaction delay applies, so that TDELAY=CDELAY(ICTYPE)=CDELAY(4).

58. SUBROUTINE ROTATE(X,Y,ANG)

This routine is called from subroutine EV12 to impose a heading error of ANG (radians) on a fighter intercept course. (X,Y) are initially the coordinates of the calculated interception point, relative to the fighter's current position. They are then rotated to correspond to the fighter's actual heading when the heading error is taken into account. If the model is in stochastic mode the error is sampled from a normal distribution with standard deviation ANG, while in deterministic mode a systematic clockwise error of ANG radians is assumed. (see Figure 58.1)

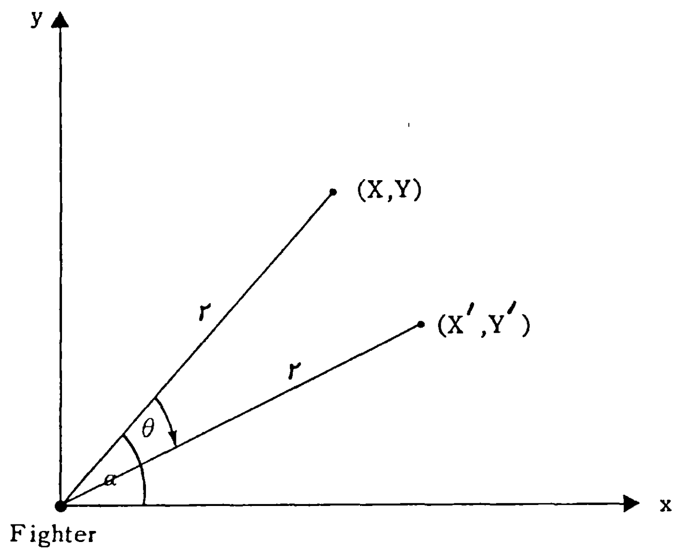


FIGURE 58.1. SUBROUTINE ROTATE

If

(X,Y) = calculated interception point

(X′,Y′) = point towards which the fighter is actually heading

and

θ = heading error

then

$$X' = r\cos(\alpha-\theta) = X\cos\theta + Y\sin\theta$$
$$Y' = r\sin(\alpha-\theta) = Y\cos\theta - X\sin\theta \qquad (58.1)$$

59. SUBROUTINE RREAD(A,N)

This subroutine is called from subroutine INPUT to read a real array A, of dimension N.

60. SUBROUTINE RTRACKS

This routine is called from subroutine INPUT to set up the track parameters for enemy raids. For each raid IR (IR=1,...,NR), the position and speed of the raid at each track-change point are specified in the input data, as is the number of raid track-change points NT(IR). (NT(IR)$\geqslant$2 since the start position of the raid is classed as the first track-change point). For each such point IT (IT=2,.....,NT(IR)) this routine determines TR(IT,IR), the time at which the raid is expected to reach the point IT, together with VXR(IT,IR) and VYR(IT,IR), the x- and y- components of the raid velocity on the track-leg from point IT-1 to point IT. The time TR(1,IR) is the time at which the raid reaches track-change point 1, ie the start time of the raid, and is specified separately in the input data.

61. SUBROUTINE SETIF(MODE1,MODE2)

This routine finds the range IF1 to IF2 of fighters IF with control mode MODEC between MODE1 and MODE2 inclusive, ie fighters IF such that:

$$1 \leqslant MODE1 \leqslant MODEC(IF) \leqslant MODE2 \leqslant 4 \qquad (61.1)$$

It is called from subroutines FFIND, MOVEF and RADAR. A simple test is made on IFM(I) for I=MODE1,.....,MODE2 to determine IF1, the serial of the first fighter in this range of control modes. If there is no such fighter then IF1=0. Otherwise a simple test is made on IFM(J), J=MODE2+1,...,5 to determine IF2, the serial of the first fighter with control mode greater than MODE2. (Note that IFM(5), the first fighter serial available for future allocation, is never zero, so that IF2 is never zero). The last fighter in the required range then has serial IDOWN(IF2).

62. SUBROUTINE SFROMT (K,INIT,V,T,DIST,FLT)

This routine calculates the distance travelled, DIST and fuel consumed, FLT by a fighter of type IFTY in time T, if it starts with initial speed V and INIT is the last profile point reached in profile K. It is called from Logical Function IDFUEL. Essentially, the distance covered on each leg of the profile is calculated until the fighter has flown for a time T. Allowances are made if the fighter starts between profile-points and if it is between profile-points after a time T. The logic of this subroutine is illustrated in Figure 62.1 and the following notes refer to details of this flowchart.

(1) The fighter speed and fuel consumption rates are assumed to be monotonically increasing so that once a fighter reaches a constant speed on a profile it remains at that speed while it remains on that profile.

(2) This resets the point (T1,V1) at which calculation of DIST and FLT begins. V1 is set equal to the current fighter speed V, while T1 is set equal to T* (see Figure 62.2), the time corresponding to speed V; this is calculated by linear interpolation.

(3) In this case the distance covered and fuel consumed while travelling from time T1 to time T2 must be included. Each section of the fighter profile is then considered in turn until a profile point T2 is reached which is greater than or equal to TSTOP.

(4) The terminal speed V* at time TSTOP is calculated by linear interpolation; for simplicity this also is illustrated in Figure 62.2. V2 is then set equal to V* and T2 set equal to TSTOP.

63. SUBROUTINE SOJPOWER

This routine is called from subroutine RADAR to evaluate the jamming power received by fighter IF from stand-off jammer IJ, when its mainbeam is illuminating raid IR. A check is first made that the jammer is switched on (SOJT(I.J) $\geqslant$ TIME) and that it does have jamming power in band IB (PSOJ(IB,IJ) $>$ 0). IB is the jamming band into which the fighter's AI frequency calls and is calculated in subroutine RADAR.

The coordinates (RELXR,RELYR) of the raid relative to the fighter are calculated in subroutine RADAR, while (DXJ,DYJ) denote the coordinates of the stand-off jammer relative to the fighter. From Figure 63.1, the angle θ subtended at the fighter by the raid and the stand-off jammer may then be calculated in subroutine DOTPROD from the expression

$$\underline{r}_R \cdot \underline{r}_J = |\underline{r}_R|\,|\underline{r}_J| \cos\theta \qquad (63.1)$$

IANGLE, the angle nearest to θ at which radar gain patterns are specified, is then obtained from Function IANG.

The power of the stand-off jammer at fighter IF, DPOWER, is calculated assuming the transmitter is isotropic:

$$\text{DPOWER} = P_J \cdot \left(L_R \cdot L_{pol} \cdot \frac{\lambda^2}{(4\pi)^2} \right) \cdot \frac{G \cdot \Delta F}{R_J^2 \cdot \Delta J} \qquad (63.2)$$

where

P_J=PSOJ(IB,IJ) = SOJ power in AI frequency band IB

L_R =1-way AI reception loss (a function of the fighter type, denoted by IFTY).

L_{pol} = Jammer polarisation loss (affects reception and transmission by all jammers equally)

λ = AI radar wavelength

and PJATR= $\frac{L_R . L_{pol} . \lambda^2}{(4\pi)^2}$ is calculated in subroutine INPUT.

also: G=GRMAX(IFTY)=AI mainbeam gain

R_J^2 = R2SOJ=squared range from SOJ to fighter

ΔF = RBW(IFTY)=AI received bandwidth

ΔJ = IBW=width of each jamming band (determined in subroutine INPUT).

The effective jamming power, P, when the fighter is looking at raid IR is then determined:

$$P=DPOWER . \frac{G_\theta}{G} \qquad (63.3)$$

where

G_θ = GR1(IANGLE,IFTY) is the AI radar gain of fighter IF in the direction θ

Assuming that the jammers do not interfere with each other, the total jamming power POWER suffered by fighter IF when looking at raid IR is finally increased by this component:

$$POWER \equiv POWER+P \qquad (63.4)$$

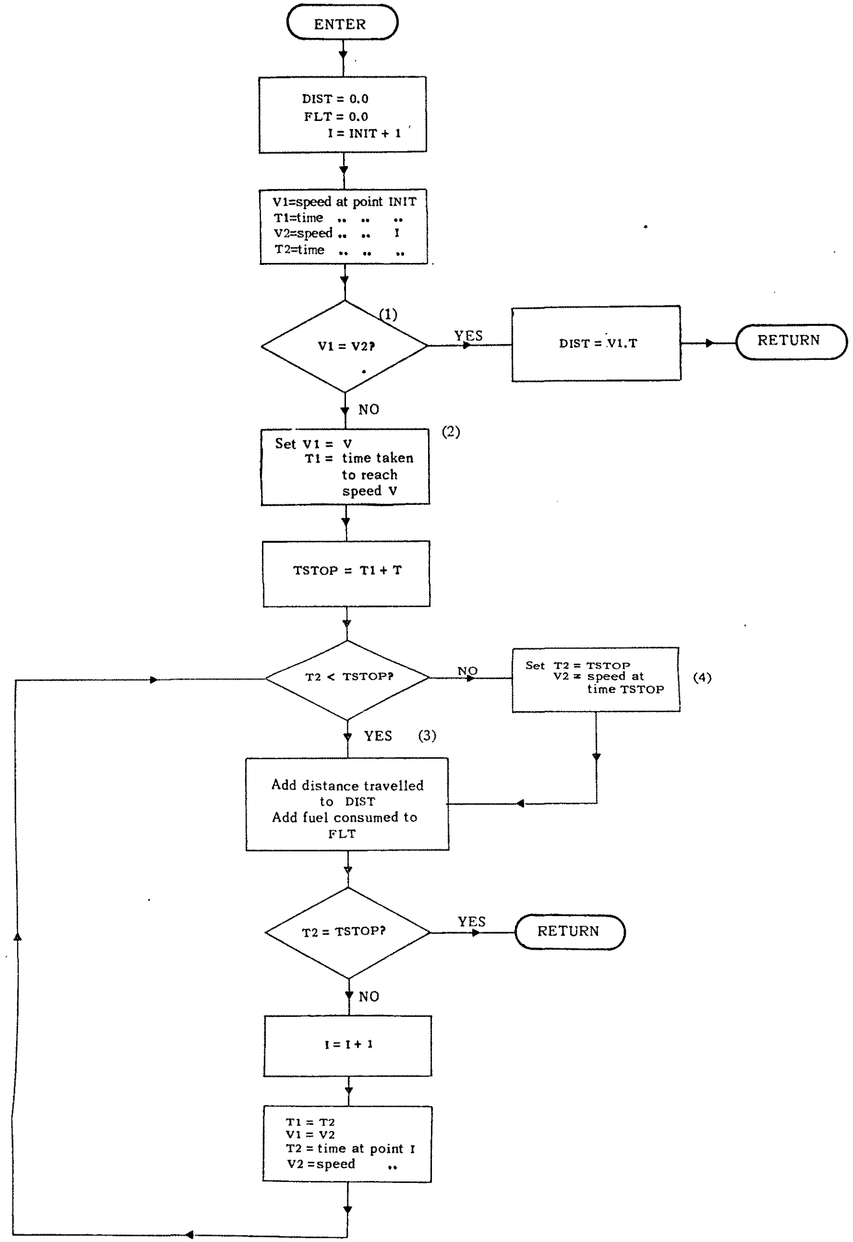


FIGURE 62.1. PROGRAM LOGIC – SUBROUTINE SFROMT

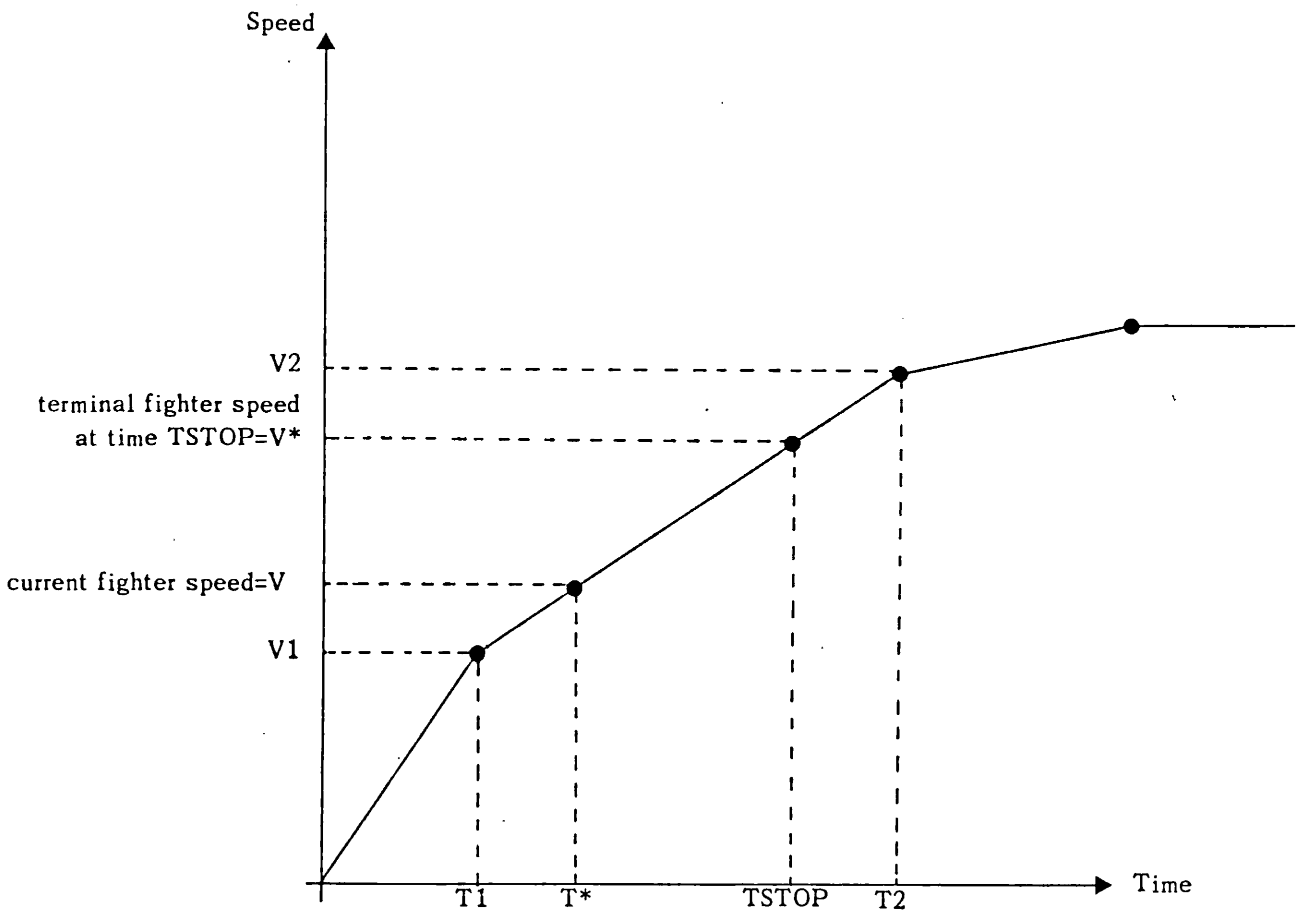


FIGURE 62.2. POINTS T* AND TSTOP AT WHICH CALCULATION BEGINS AND ENDS – SUBROUTINE SFROMT

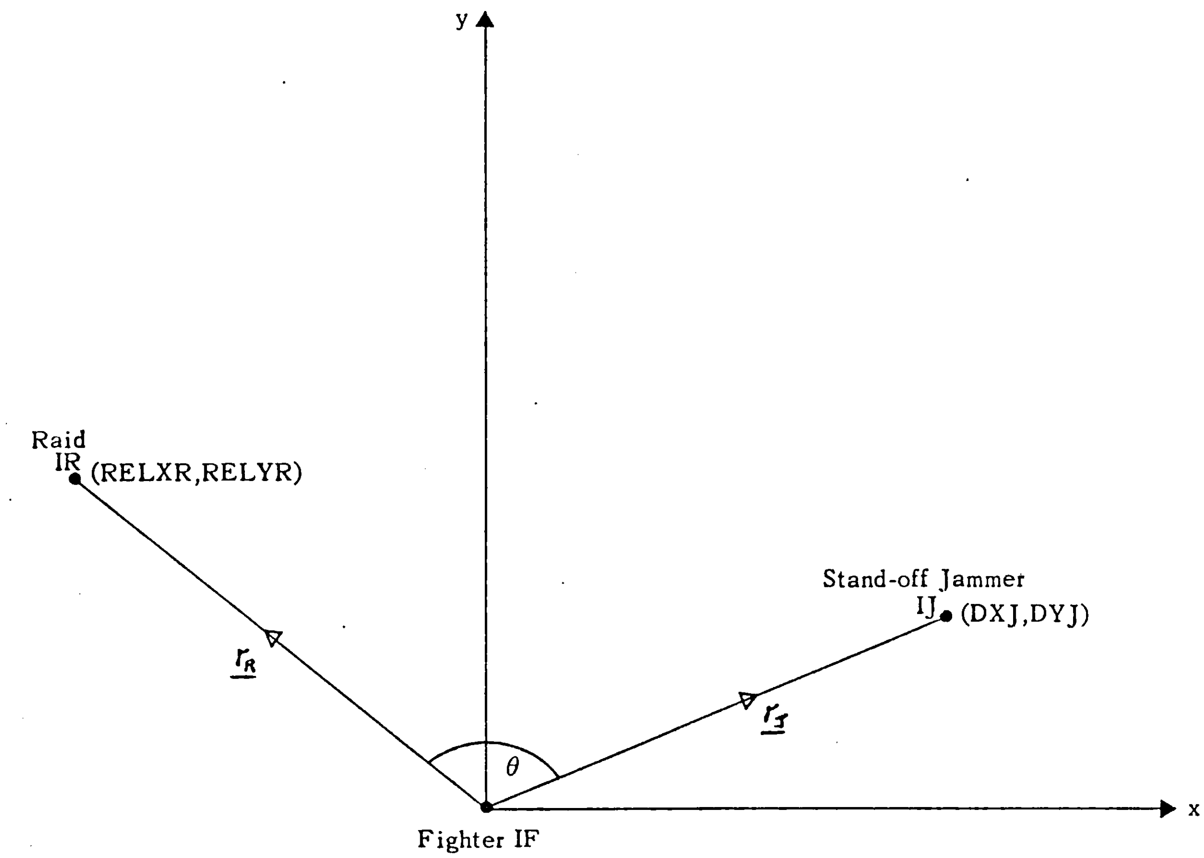


FIGURE 63.1. FIGHTER, RAID AND STAND-OFF JAMMER GEOMETRY – SUBROUTINE SOJPOWER

64. SUBROUTINE STATS

This routine is called from subroutines RADEVENTS,LAUNCHTEST AND EVENTS at event types 12 (collision) and 16 (missile-splash) to collect statistical data. For each fighter, data is aggregated on the times at which detection, burnthrough and missile-launch occur together with the time at which collision or missile-splash occurs and the distance at that point of the raid from its target. Aggregated values of the squares of these random variables are also calculated so that their standard deviations may be derived in subroutine OUTPUT, together with their mean, minimum and maximum values.

65. SUBROUTINE STATSINIT

This routine is called from subroutine INPUT at the beginning of a model run to initialise all the statistical variables used in subroutines STATS and OUTPUT.

66. SUBROUTINE TESTEVENTS

This routine is called from subroutine RADEVENTS if fighter IF has acquired range on its target, raid IR. It tests whether the fighter is sufficiently close to the raid to resolve the raid size (ie the approximate number of aircraft in the raid), if this has not already occurred as a result of the raid crossing warning line 3 (in which case the variable IEV5(IR) is set equal to 1). The test currently used is based only on the angular resolution of the raid, as illustrated in Figure 66.1. It may be worthwhile to eventually introduce tests based on range or velocity resolution.

The range RR from the fighter to the raid and the angle ϕ illustrated in Figure 66.1 are calculated in subroutine RADAR, where ϕ is stored in the array element PHI(IF,IR). The width of the fighter's radar beam when it illuminates the raid is then

$$\text{DRADAR=RR.BEAMW} \qquad (66.1)$$

where BEAMW is the AI 3-dB beamwidth, specified in the input data. Only if this width, DRADAR, is less than the distance

$$\text{DRAID}=|y\cos\phi| + x\sin\phi \qquad (66.2)$$

is the fighter deemed to have resolved the raid size (y is the inter-aircraft spacing, x is the inter-flight spacing of the raid), in which case an event type 5 for this raid is generated a time TRESPONSEF hence. TRESPONSEF, specified in the input data for each fighter type, is the time taken for Ground Control to absorb and process the fighter's information .

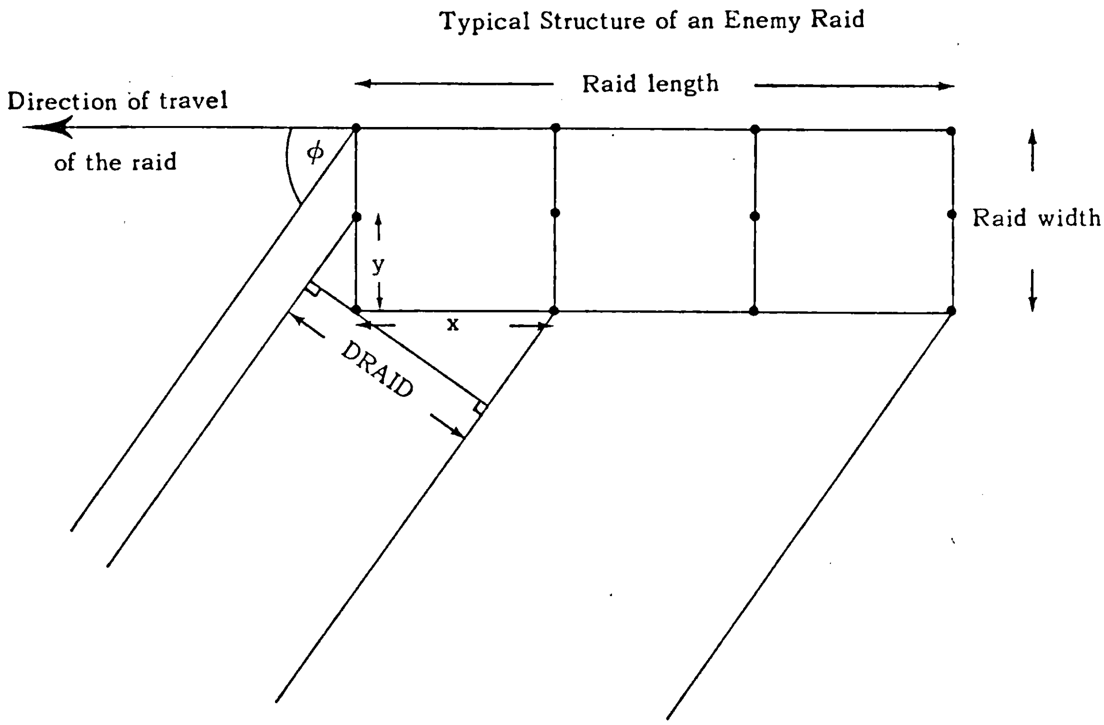


FIGURE 66.1. RESOLUTION OF RAID SIZE – SUBROUTINE TESTEVENTS

y = inter-aircraft spacing = SPLAT(IR) (specified in the input data)

x = inter-flight spacing = STRAIL(IR) (.. .. ..)

$$DRAID = |y\cos\phi| + x\sin\phi$$

67. SUBROUTINE TESTFIGHTER

This routine is called from subroutine RADAR to determine if any further radar and ECM processing is required for fighter IF. Such processing is unnecessary in the following circumstances:

(i) The fighter has not yet taken off (MODER(IF)=0)

(ii) If event type 12 (collision) rather than type 16 (missile-splash) is the measure of fighter effectiveness, so that I12OR16=1, when the fighter acquires range on its target and adopts its own interception course the variable ICOLLIDE(IF) is set equal to 1 in subroutine RADEVENTS. No further radar processing of this fighter is carried out before its planned collision.

(iii) If the fighter has completed an attack and is turning to continue its reattack sequence, the variable ITURN(IF) is set equal to 1 in subroutine REATTACK. The radar processing of this fighter is suspended until it has completed its turn, represented by an event type 18.

68. SUBROUTINE TFROMS(K,INIT,IFTY,V,T,DIST,FLT)

This subroutine calculates the time taken, T, and the fuel consumed,FLT, by a fighter of type IFTY at profile point INIT on profile K in travelling a distance DIST, if it starts with initial speed V. It is called from subroutinesEV9,INPUT and EVENTS at event types 9 and 10. Basically, the time taken to cover each leg of the profile is calculated until the distance DIST has been covered. Adjustments are necessary since the fighter is normally between profile-points at entry to this routine and is between profile-points after a distance DIST has been covered. The subroutine logic is illustrated in Figure 68.1 and the following notes refer to details of this flowchart.

(1) The fighter speed and fuel consumption rates are assumed to be monotonically increasing, so that once a fighter reaches a constant speed on a profile it remains at that speed while it remains on that profile.

(2) This resets the point (T1,V1) on the fighter profile at which calculation of T and FLT begins. As in Figures 62.1 and 62.2 for subroutine SFROMT,V1 is set equal to the initial fighter speed, V, while T1 is set equal to T*, the time corresponding to this speed; this is calculated by linear interpolation.

(3) STEMP denotes the distance travelled from time T1 to T2.

(4) In this case the fighter reaches at least profile point (T2,V2) before the distance DIST is covered, so that T is increased by (T2-T1), the variable S which denotes the total distance covered up to profile point (T2,V2) is increased, and the next leg of the fighter profile is considered.

(5) The acceleration F from T1 to T2 is:

$$F=\frac{(V2-V1)}{(T2-T1)} \qquad (68.1)$$

The fighter's terminal speed V* when it has travelled the required distance, DIST, is given by

$$(V^*)^2 = (V1)^2 + 2.F.(DIST-S) \qquad (68.2)$$

The time, TEXTRA, spent on this profile leg is then

$$TEXTRA = \frac{2.(DIST-S)}{(V1+V2)} \qquad (68.3)$$

Finally, the time TEXTRA is added to T to give the total time taken to cover the distance DIST and the fuel consumed while doing so, FLT, is also calculated.

69. SUBROUTINE TIMETOTS(TOTMEAN,TMEAN,SDTOT,VAR, TOTMIN,TMIN,TOTMAX,TMAX,N)

This subroutine is called for each fighter IF from subroutine OUTPUT. It increases the statistical variables TOTMEAN,SDTOT,TOTMIN and TOTMAX by addition of TMEAN, VAR,TMIN and TMAX respectively, while N is increased by 1. It generates the totals from which can be calculated the mean, variance, minimum and maximum values summed over all fighters and over all replications of a number of random variables. N is a count of the number of times the subroutine is called for each random variable. After this subroutine has been called for every fighter, the final totals obtained are used as inputs to subroutine MEANTIMES.

70. SUBROUTINE UPDATEERROR(IR,I)

The routine updates the standard deviations of the errors which are applied to estimates of the parameters of raid IR, together with the reaction delay appropriate to a track-change by this raid. Errors and reaction delay depend upon the source of information - Ground Control (GC) or fighter - and the source providing the smallest errors or shortest delays is assumed to be chosen. If a conflict arises between the sources of information providing the smallest errors and the shortest reaction delay, a criterion must be specified in the model to determine which of these factors takes precedence.

The variable I in the parameter list equals 0 if this routine is called when raid track information first becomes available from GC sources, ie when the raid crosses warning line 2, generating an event type 4. Subroutine UPDATERROR can also be called from subroutine RADEVENTS, if a fighter of type I achieves burnthrough against raid IR.

TRESPONSER(IR) equals the current track-change reaction delay appertaining to raid IR;depending on the best available source of information, it either equals TRESPONSEF(I), the delay generated by fighter type I, or GROUNDELAY1, the GC reaction delay if the raid has crossed warning line 2. RPOSERR(IR),RSPEEDERR(IR) and RDIRERR(IR) are, respectively, the standard deviations of the current errors in estimates of the raid position, speed and heading. Depending on the best available source of information, these correspond to fighter information - FPOSERR(I),FSPEEDERR(I) and FDIRERR(I) - or GC generated information, viz. GROUNDERROR, GSPEEDERROR and GDIRERROR respectively.

71. SUBROUTINE WRITERAD(I)

This subroutine is called from subroutine RADEVENTS to print information if a fighter detects a raid, either in the clear (I=1) or as a jammer (I=2), or if a fighter acquires range information on a raid (I=3). It is also called from subroutine LAUNCHTEST with I=4 to record fighter and raid positions and velocities at missile-launch. Finally, it is called from subroutine EVENTS, with I=5, to record this information if an event type 12 (a fighter reaches its expected interception point) or type 16 (missile-splash) occurs; in these cases the distance of the raid from its target is also calculated and printed.

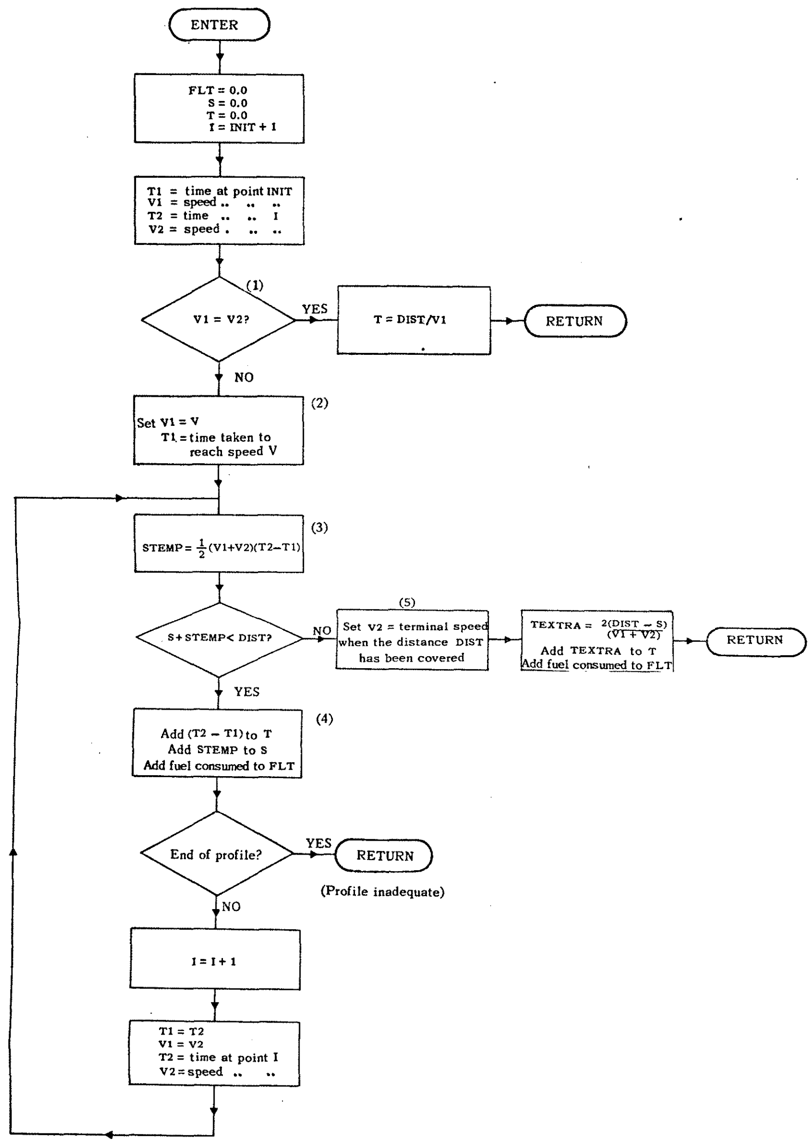


FIGURE 68.1. PROGRAM LOGIC – SUBROUTINE TFROMS

72. PULSE DOPPLER CLUTTER

A fairly general treatment of pulse doppler clutter is presented in this section, including the effects of non-level flight, a linear FM ranging modulation and a spherical earth. Any particular airborne radar may not exhibit all of these complications, but an appropriate modification of the general case can easily be obtained by setting various parameters to zero. Subroutine ZCLUTTERCALC, with subsidiary routines, contains the processing corresponding to this theory.

The choice of coordinate system considerably simplifies the analysis (see Figure 72.1). The radar is at the point R with coordinates (0,0,h), ie directly above the chosen origin of coordinates. The radar velocity is $\underline{V}=(V_x, 0, V_z)$, ie it moves in the x-z plane. P is the position of a general clutter point on the earth's surface.

Let $\underline{r}$ denote the vector from R to P, with

$$r = |\underline{r}| \qquad (72.1)$$

If R_e denotes the effective earth radius, then from triangle CPR in Figure 72.2 the cosine rule gives

$$r^2 = x^2 + y^2 + \left(h + \frac{r^2 - h^2}{2(R_e + h)}\right)^2 \qquad (72.2)$$

The doppler frequency shift, f, of the signal returned from the target is known and is approximated by

$$f = \frac{2.V_c}{\lambda} \qquad (72.3)$$

where V_C is the relative velocity of the target with respect to the radar and λ is the radar carrier wavelength.

Pulse doppler clutter is returned from those points on the ground which return the radar signal from main beam and sidelobes with the same doppler frequency, f, as the target. The condition for the point P to produce a pulse doppler clutter return at the doppler frequency f is given by

$$f = \frac{2\underline{v}.\underline{r}}{\lambda r} + \frac{2\dot{f}r}{c} \qquad (72.4)$$

(It is assumed in subroutine ZCLUTTERCALC that the radar has a triple linear ranging modulation, $\dot{f}$ - see Figure 74.1. The program calculates the clutter for each of the three regions of the carrier frequency modulation shown ie AB,BC and CA, and averages the three results obtained).

The clutter power dP received from a clutter patch of elemental area dA at the point P is then given by:

$$dP = \left(\frac{P.G}{4\pi r^2}\right).\left(\frac{\sigma_0 \, dA}{4\pi r^2}\right).\left(\frac{G\lambda^2}{4\pi}\right) \tag{72.5}$$

↑ effective area of the radar antenna

ie

$$dP = \frac{P\lambda^2 \sigma_0 G^2 dA}{(4\pi)^3 r^4} \tag{72.6}$$

where:

P = mean transmitter power (including the effects of all losses; to represent the effects of eclipsing by an average loss factor entails the assumption that r is much larger than the inter-pulse distance, which is normally true for pulse-doppler radar).

and

$\sigma_0(x,y)$ = ground scattering cross-section / sq.m. at the point $\underline{r}$

$G(x,y)$ = radar antenna gain in the direction $\underline{r}$

From Figure 72.2 it can be seen that

$$dA = \frac{dx.dy}{\cos\phi} \tag{72.7}$$

From the cosine rule in triangle CPR we have

$$\cos\phi = 1 - \frac{r^2 - h^2}{2R_e(R_e + h)} \tag{72.8}$$

To integrate make the change of variables $(x,y) \longrightarrow (r^2, f)$, which has the Jacobian

$$J = \frac{4V_x.|y|}{\lambda r W} \tag{72.9}$$

where

$$W = 1 - \frac{h + \dfrac{r^2 - h^2}{2(R_e + h)}}{R_e + h} \tag{72.10}$$

Now, from the result:

$$dr^2 . df = |J| \, dx . dy \tag{72.11}$$

equations (72.6) and (72.9) give:

$$dP = \frac{P\lambda^3 G^2 \sigma_0 W \, dr^2 \, df}{4(4\pi)^3 |V_x| r^3 \cos\varphi . |y|} \tag{72.12}$$

The appropriate range of r^2 in the integration is from $r^2 = h^2$ to $r^2 = 2hR_e + h^2$. The range of integration is from r=h until the clutter patch is exactly at the radar horizon; this occurs when the grazing angle θ_g is zero. The cosine rule in Figure 72.2 gives

$$\sin\theta_g = \frac{2hR_e + h^2 - r^2}{2rR_e} \tag{72.13}$$

so if $\theta_g = 0$ then $r^2 = 2hR_e + h^2$.

Integrating over r^2 and omitting the differential factor df to leave power per unit bandwidth, denoted by P(f), equation (72.12) gives

$$P(f) = \frac{P\lambda^3}{4(4\pi)^3 |V_x|} \cdot \int_{r^2 = h^2}^{r^2 = h^2 + 2hR_e} \left(\frac{G^2 \sigma_0 W}{r^3 \cos\varphi \, |y|} \right) dr^2 \tag{72.14}$$

The total clutter power, PC, is then simply taken to be

$$PC = P(f) . \Delta f \tag{72.15}$$

where Δf is the effective bandwidth of the pulse doppler radar (assumed to be of the order of a few hundred Hz).

The integrand of (72.14) is not straightforward. To evaluate it the integral and all parameters are first made dimensionless by expressing all lengths in terms of h, so that we write:

$$X = \frac{x}{h}$$

$$Y = \frac{y}{h}$$

$$Z = \frac{z}{h} \qquad (72.16)$$

and

$$R = \frac{r}{h}$$

Equation (72.14) becomes:

$$P(f) = \frac{P\lambda^3}{4(4\pi)^3 |V_x| h^2} \cdot \int_1^{1+\frac{2R_e}{h}} \frac{G^2 \sigma_0 W}{R^3 \cos\phi . |Y|} \, dR^2 \qquad (72.17)$$

where, from (72.10):

$$W = 1 - \frac{1 + \frac{R^2 - 1}{2(1 + R_e/h)}}{\left(1 + \frac{R_e}{h}\right)} \qquad 72.18)$$

and from (72.8):

$$\cos\phi = 1 - \frac{R^2 - 1}{2\frac{R_e}{h}\left(1 + \frac{R_e}{h}\right)} \qquad (72.19)$$

The theoretical expression used for σ_0 is:

$$\sigma_0 = \sigma_{00} (\sin \theta_g)^{\alpha} \qquad (72.20)$$

where θ_g is the grazing angle and σ_{oo} is the value of σ_o at normal incidence; the values of the constants σ_{oo} and α, for each radar type, must be specified in the program input data. From equation (72.13):

$$\sin\theta_g = \frac{1 + \frac{2R_e}{h} - R^2}{2R.\frac{R_e}{h}} \tag{72.21}$$

Now, for a particular value of r^2 the x-coordinate of any clutter point corresponding to this value is derived from (72.4), which may be written in the following expanded form:

$$x = \frac{\lambda f r}{2V_x} - \frac{\lambda \dot{f} r^2}{cV_x} + \left(h + \frac{r^2 - h^2}{2(R_e + h)}\right) \cdot \frac{V_z}{V_x} \tag{72.22}$$

Equivalently, for a particular value of R^2 the corresponding value of X is given by:

$$X = \frac{\lambda f R}{2V_x} - \frac{\lambda \dot{f} h R^2}{cV_x} + \left(1 + \frac{R^2 - 1}{2\left(1 + \frac{R_e}{h}\right)}\right) \cdot \frac{V_z}{V_x} \tag{72.23}$$

This may be written:

$$X = A1.R + A2.R^2 + HH.B \tag{72.24}$$

where

$$A1 = \frac{\lambda f}{2V_x}$$

$$A2 = \frac{-\lambda \dot{f} h}{cV_x} \tag{72.25}$$

$$HH = 1 + \frac{R^2 - 1}{2\left(1 + \frac{R_e}{h}\right)}$$

$$B = \frac{V_z}{V_x}$$

(A1,A2,HH and B are actual program variables).

Finally, the corresponding value of Y^2 can be derived from equation (72.2):

$$X^2 + Y^2 + (HH)^2 = R^2 \tag{72.26}$$

If the resulting value of Y^2 is negative, the corresponding value of R^2 does not represent a valid clutter return point, and does not therefore contribute to the integral. If on the other hand Y^2 turns out to be positive, Y can be evaluated and the points corresponding to both +Y and -Y contribute to the integral. G^2 is in fact the only part of the integral which is affected by the sign of Y, and the two values are expressed most simply by rewriting G^2 as

$$G^2(X,Y,Z)+G^2(X,-Y,Z)$$

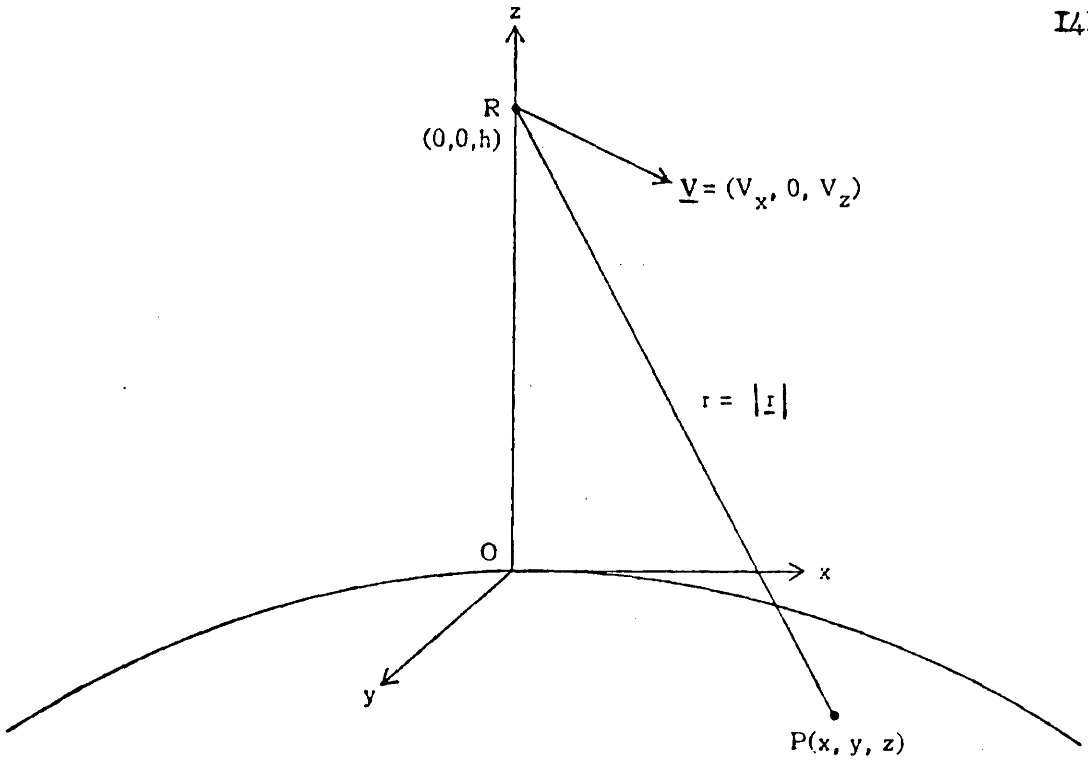


FIGURE 72.1. CHOICE OF COORDINATE SYSTEM, PULSE DOPPLER CLUTTER

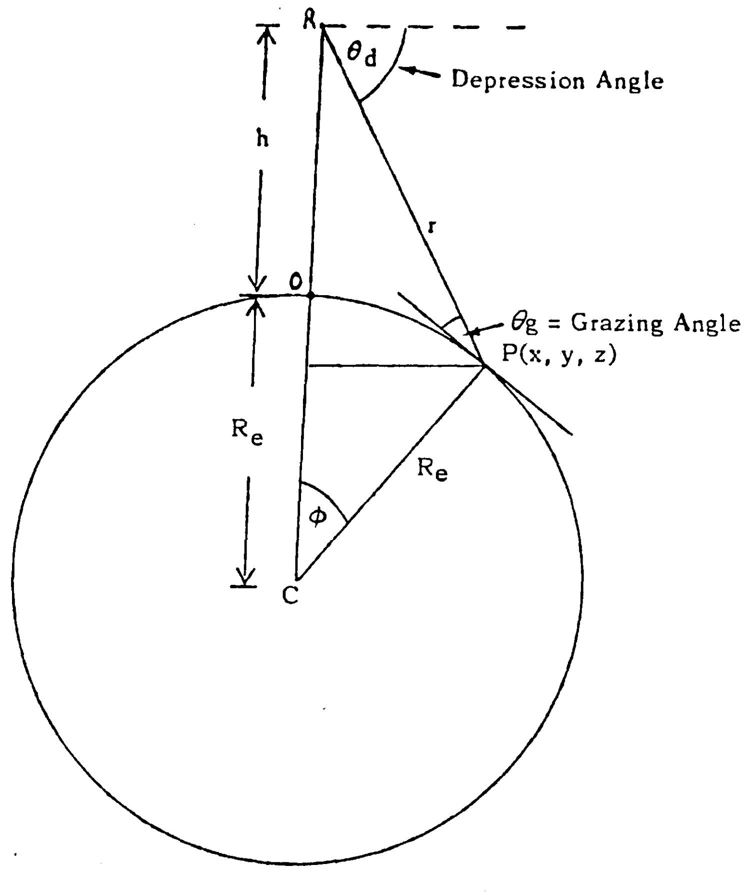


FIGURE 72.2. GEOMETRY OF RADAR, R, AND PULSE DOPPLER CLUTTER POINT, P

73. SUBROUTINE ZCLUTTERCALC(PDCL)

This routine is called from subroutine RADAR to calculate the pulse doppler clutter of a fighter in the direction of a target, when its mainbeam is illuminating that target. The processing in this routine is based on the theory derived in the previous section, and the same notation is used here.

It is required to calculate the following integral:

$$PDCL = \int_{1}^{1+\frac{2R_e}{h}} \frac{G^2 \sigma_0 W}{R^3 \cos\phi . |Y|} \, dR^2 \qquad (73.1)$$

The integration is performed numerically, where $|Y|$ is evaluated as $\sqrt{Y^2}$ provided Y^2 is positive. Integration over the whole interval $[1, 1+2R_e/h]$ may fail because of singularities at $Y^2=0$. Hence the integration range is first searched for roots of $Y^2=0$, so that the integration may be performed only over those sub-intervals on which Y^2 is positive. A final manipulation of the integral is necessary to eliminate singularities at the end-points of each such sub-interval.

Figure (73.1) illustrates the logic of this subroutine and the following notes refer to details of this flowchart.

(1) The clutter routines have been written independently of any particular application to the fighter model. Although it is not shown here, subroutine ZCLUTTERINIT is called immediately subroutine ZCLUTTERCALC is entered. This routine assigns values to clutter parameters which are fighter or raid dependent - fighter velocity components, doppler frequency shift of the signal returned from the target, etc.

(2) The following variables are calculated:

$$TY = 1 - TX^2 - TZ^2$$

(the y-direction cosine of the target from the radar)

$$B = \frac{VZ}{VX} = \frac{V_z}{V_x}$$

$$A1 = \frac{\lambda f}{2V_x}$$

$$D = \frac{2R_e}{h} \qquad (73.2)$$

$$UPLIM = 1 + \frac{2R_e}{h}$$

(the upper limit of integration)

$$C = \frac{1}{2 + \frac{2R_e}{h}}$$

POWER = 0.0

(the total clutter power; initially set equal to zero)

(3) The carrier frequency ranging modulation - if any - is assumed to be triangular, as shown in Figure 74.1. If the slope $\dot{f}$ is non-zero, the clutter integral is calculated over each of the three regions - zero modulation, modulation with slope $+\dot{f}$ and modulation with slope $-\dot{f}$ - and the average of the three results is taken.

(4) The following variables are calculated:

$$FDOT1 = \dot{f}.(J-2)$$

$$A2 = -\frac{\lambda . FDOT1}{c.V_x} \qquad (73.3)$$

(see equation (72.25)

NROOTS is a count of the number of roots of $Y^2=0$ found in the integration range, so it is initially set equal to zero. CLUTTER is a logical variable which is set to TRUE when a valid clutter return point is first found; it is initially set to FALSE.

(5) This section of the routine is described fully later - see Figure (73.2). If there are roots of $Y^2=0$ in the integration range these are stored in the array ROOT, while the corresponding slope $\frac{\partial Y^2}{\partial R^2}$ at these points is stored in the array SLOPE.

(6) If there are roots of $Y^2=0$ in the integration range a simple check is carried out that a root has not been missed, viz that the slopes of successive roots alternate in sign. Note that, when $R^2=1$, Y^2 is negative or zero, since at that point from (72.25):

$$HH = 1 \tag{73.4}$$

while from (72.26):

$$X^2 + Y^2 = 0 \tag{73.5}$$

(7) Each root is now considered in turn to derive valid integration regions for R^2. Each root, RVALUE, and its corresponding slope, GRAD, is in turn recovered from the ROOT and SLOPE arrays respectively:

$$\left.\begin{array}{l} \text{RVALUE} = \text{ROOT (IROOT)} \\ \\ \text{GRAD} \quad = \text{SLOPE(IROOT)} \end{array}\right) \quad \text{IROOT=1,....., NROOTS} \tag{73.6}$$

(8) The procedure is to split up each valid integration region of R^2 so that there is a singularity in the integrand at only <u>one</u> end - point; see Figure 73.3. In particular, on first entry through this loop GRAD should be positive.

(9) The lower limit of integration is X_o, where

$$X_o = \text{RVALUE} \tag{73.7}$$

The upper integration limit, X_1, is provisionally set equal to UPLIM. If there are any further roots remaining, the required integration region is of type ①, as shown in Figure 73.3. The upper limit X_1 is then given by:

$$\frac{1}{X_1} = \frac{1}{2}\left(\frac{1}{X_0} + \frac{1}{X_2}\right) \tag{73.8}$$

This expression is used rather than simply $\frac{1}{2}(X_0+X_2)$ because, in subroutine GINT, the variable of integration is changed from R^2 to $\frac{1}{R^2}$.

(10) In this case the integration region is of type (2), as shown in Figure 73.3. The lower integration limit, X_0, is set equal to the previous value of X_1, while X_1 is set equal to RVALUE:

$$X_0 = X_1$$
$$X_1 = \text{RVALUE} \qquad (73.9)$$

(11) This routine evaluates the following integral by a Gaussian algorithm:

$$\text{DPOWER} = \int_{X_0}^{X_1} \frac{G^2 \sigma_0 W}{R^3 \cos\phi . |Y|} \, dR^2 \qquad (73.10)$$

(12) The total clutter power, POWER, is incremented by the contribution DPOWER from the sub-interval $[X_0, X_1]$.

(13) If there are no roots of $Y^2=0$ in the integration region [1,UPLIM] and CLUTTER still equals FALSE, there is no clutter generated anywhere in the integration region.

(14) If CLUTTER=TRUE, clutter is generated over the whole region so that, in the call to subroutine ZGINT:

$$X_0 = 1.0$$
$$X_1 = \text{UPLIM} \qquad (73.11)$$

Also the variable RVALUE is set equal to -1. This enables subroutine ZGINT to avoid unnecessary manipulation of the integrand in (73.10), for there is no singularity at either end-point.

(15) Finally, the value of the integral in (73.1) is returned to subroutine RADAR in the variable PDCL. If $\dot{f}=0$ then

PDCL=POWER

while if $\dot{f} \neq 0$ then

$$\text{PDCL} = \frac{\text{POWER}}{3}$$

(see section (3)).

Calculation of Roots of $Y^2=0$

The integration range $[1, 1+\frac{2R_e}{h}]$ of R^2 is searched for roots of $Y^2=0$, where Y^2 is expressed as a function of R^2 by equation (72.26):

$$X^2 + Y^2 + (HH)^2 = R^2$$

The sign of Y^2 is examined at the points

$$(R^2)_i = \left(1+\frac{2R_e}{h}\right)^{\frac{i}{\text{NSEARCH}}} \qquad i=0,1,\dots,\text{NSEARCH} \qquad (73.12)$$

where NSEARCH is specified in the BLOCK DATA segment and currently

NSEARCH=100

(The search for roots of Y^2 is made at geometrically-spaced points rather than evenly-spaced points because initial roots tend to be closer together).

The search algorithm is illustrated in Figure 73.2. Some program variables are first initialised - the number of roots, NROOTS, is set to zero while the logical variable CLUTTER is set to FALSE; this is set to TRUE when a clutter point is found, ie a point R^2 such that the corresponding Y^2 is positive. The variable R2, which denotes the variable of integration, R^2, is set equal to 1.0, the left-hand end of the integration range; subroutine EVALUATE calculates the corresponding value of Y^2, denoted by Y2. Finally the logical variable CLUTTER is updated and the working variable YR is initialised:

$$\text{YR}=\text{Y2} \qquad (73.13)$$

Each interval $[(R^2)_{i-1},(R^2)_i]$,i=1,2...., NSEARCH is then considered in turn: YL denotes the value of Y^2 at $(R^2)_{i-1}$ and YR denotes the value of Y^2 at $(R^2)_i$. The variable CLUTTER is updated and the sign of YL.YR is examined. It is assumed that the parameter NSEARCH is sufficiently large such that, if YL.YR$>$0, there is no root of Y^2=0 in the interval $[(R^2)_{i-1},(R^2)_i]$ and the next interval is considered. Similarly, if YL.YR $<$ 0, it is assumed that there is just one root of Y^2=0 in this interval. (In fact a simple test is carried out later to check whether a root has been missed).

If in an interval YL.YR$<$0, an initial approximation to the root is found by binary chop, to within a specified accuracy E (defined in the BLOCK DATA segment). This approximation is then carried into subroutine ITERATE, where a Newton-Raphson iteration refines it still further. Subroutine ITERATE also increases the count of the number of roots found, NROOTS, and stores in the arrays ROOT and SLOPE respectively the value of the root and the corresponding value of $\partial Y^2/\partial R^2$ at this point. The next interval is then considered and the process is repeated until the whole integration range $[1,1+\frac{2R_e}{h}]$ has been examined.

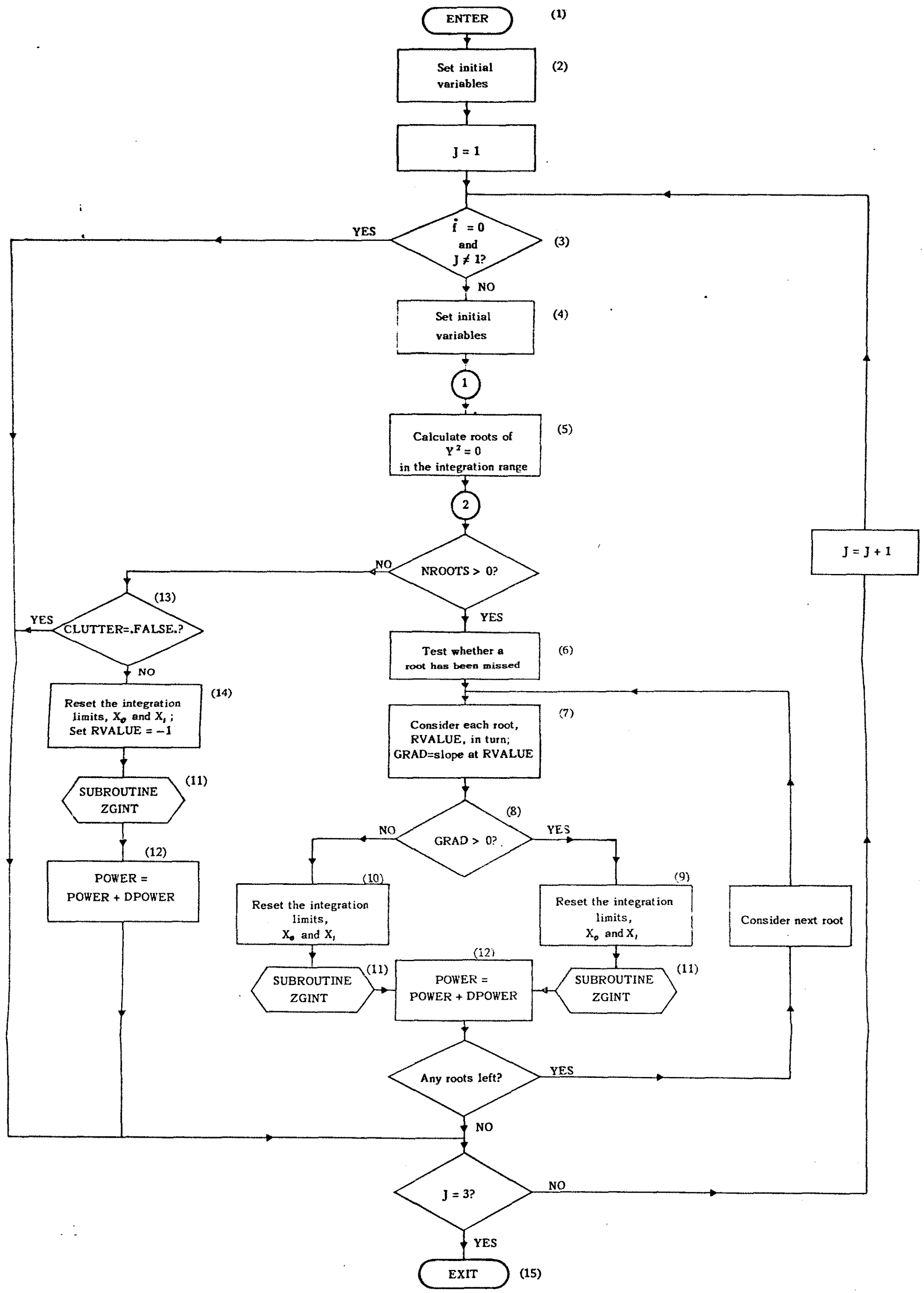


FIGURE 73.1. PROGRAM LOGIC – SUBROUTINE ZCLUTTERCALC

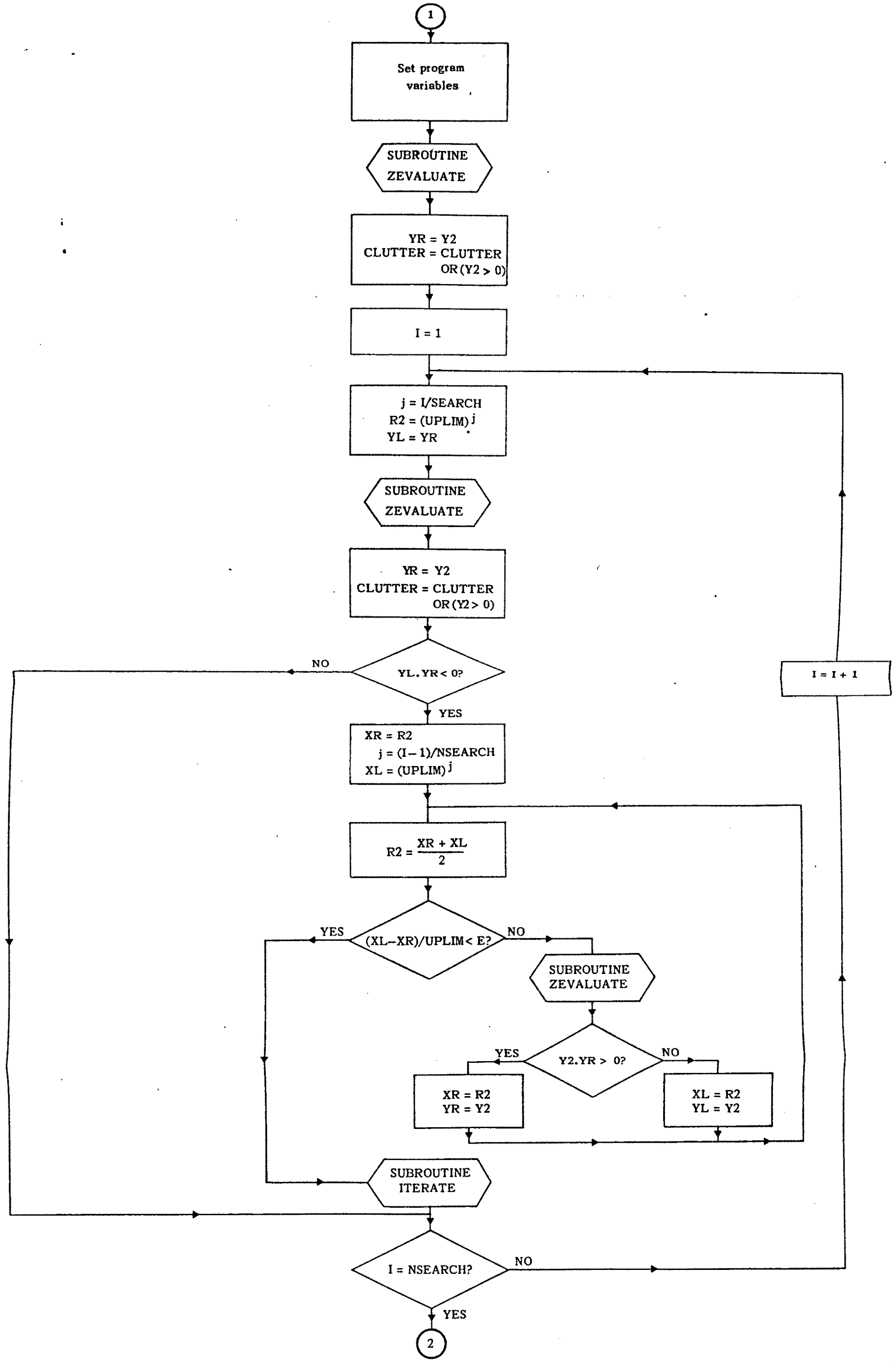


FIGURE 73.2. CALCULATION OF THE ROOTS OF $Y^2 = 0$ – SUBROUTINE ZCLUTTERCALC

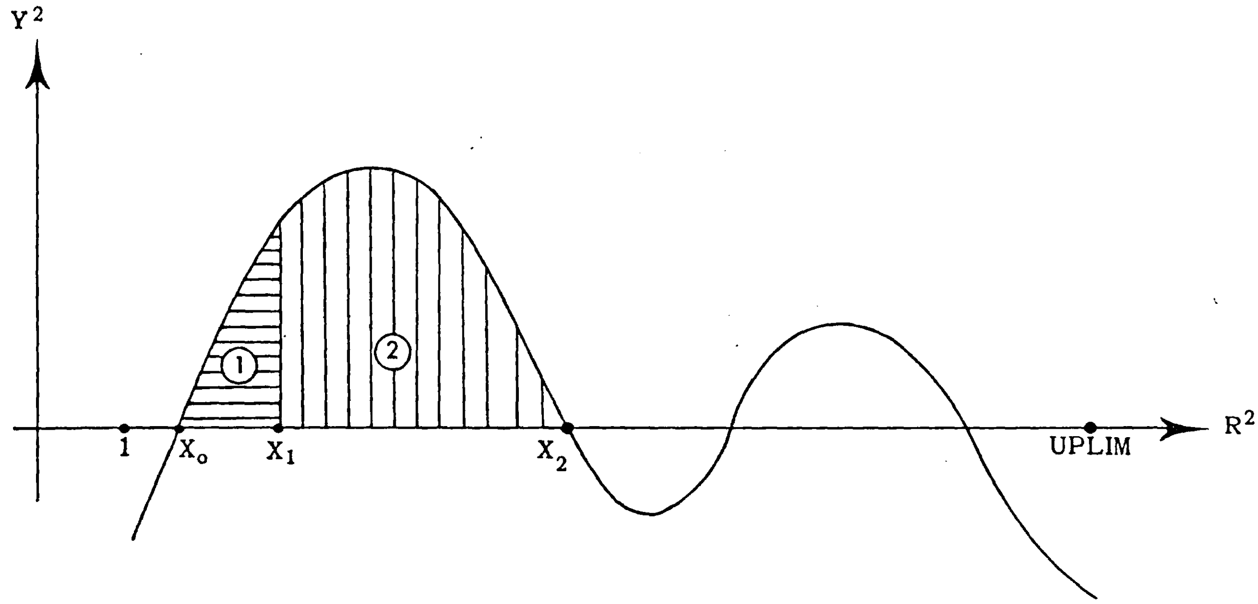


FIGURE 73.3. TWO TYPES OF INTEGRATION REGION – SUBROUTINE ZCLUTTERCALC

74. SUBROUTINE ZCLUTTERINIT

This routine is called from subroutine ZCLUTTERCALC, for fighter IF illuminating raid IR on its AI radar. It calculates some parameters used later in subroutine ZCLUTTERCALC to evaluate the fighter's clutter interference when looking at this raid. Currently the AI radar is assumed to operate in pulse doppler (PD) mode, although pulse clutter calculation routines are also incorporated in the model. The following are the parameters which are calculated:

(i) VX,VZ

These are the horizontal and vertical components respectively of the fighter's velocity. The fighter's movement in the vertical plane is currently not represented, so that

$$VZ = 0$$

and

$$VX=FV(IF) \tag{74.1}$$

Note that, from Figure 72.1, VX is the x-component of the fighter's velocity, denoted by Vx. The coordinate system adopted for the pulse doppler clutter calculations is defined by the requirement that the fighter moves in the x-z plane, so that $V_y=0$.

(ii) TX,TZ

These are the x- and z- direction cosines of the target, raid IR, with respect to the fighter. The y-direction cosine is then immediately available from

$$TX^2+TY^2+TZ^2=1 \tag{74.2}$$

Until the fighter's altitude is updated regularly in the model its attack is assumed to be at co-altitude with the raid, so that

$$TZ=0.0 \tag{74.3}$$

The coordinates (RELXR,RELYR) of the raid relative to the fighter are known, as is its range RR from the fighter, so that

$$TX = \frac{RELXR}{RR} \tag{74.4}$$

(iii) F

This is the droppler frequency shift of the signel returned from the target, and is approximated by

$$F = \frac{2V_c}{\lambda} = \frac{2.VC}{WLENGTH} \tag{74.5}$$

where

V_c =VC is the relative velocity of the target with respect to the fighter; if (RVX,RVY) and FVX,FVY) are the components of the target and fighter velocities respectively, then:

$$VC^2 = (RVX-FVX)^2+(RVY-FVY)^2 \qquad (74.6)$$

Also

λ = WLENGTH is the AI carrier wavelength

(iv) H

In subroutine ZCLUTTERCALC this denotes the AI radar altitude, currently assumed to be equal to the raid altitude:

$$H=RH(IR) \qquad (74.7)$$

(v) FDOT = $\dot{F}$

This is the slope of the AI radar linear ranging modulation on the carrier frequency in Hz/sec. If the fighter's PD radar does not possess such a facility then FDOT=0.

It is assumed that the modulation of the carrier frequency, if any, can be linearly approximated, as shown in Figure 74.1.

(vi) SIGØ,SIGEXP

The ground scattering cross-section/unit area at the clutter patch, denoted by σ_0, is assumed given by the theoretical expression:

$$\sigma_0 = \sigma_{00} \left(\sin \theta_g\right)^{\alpha} \qquad (74.8)$$

where θ_g is the grazing angle at the clutter patch (see Figure 72.2) and σ_{00} is the value of σ_0 at normal incidence ($\theta_g = \pi/2$). The values of the constants σ_{00} = SIGØ and α=SIGEXP are specified in the input data for each radar type.

(vii) GAIN

The GAIN array contains G^2, where G is the AI radar 1-way gain pattern; this is the function of G most commonly used in clutter calculations.

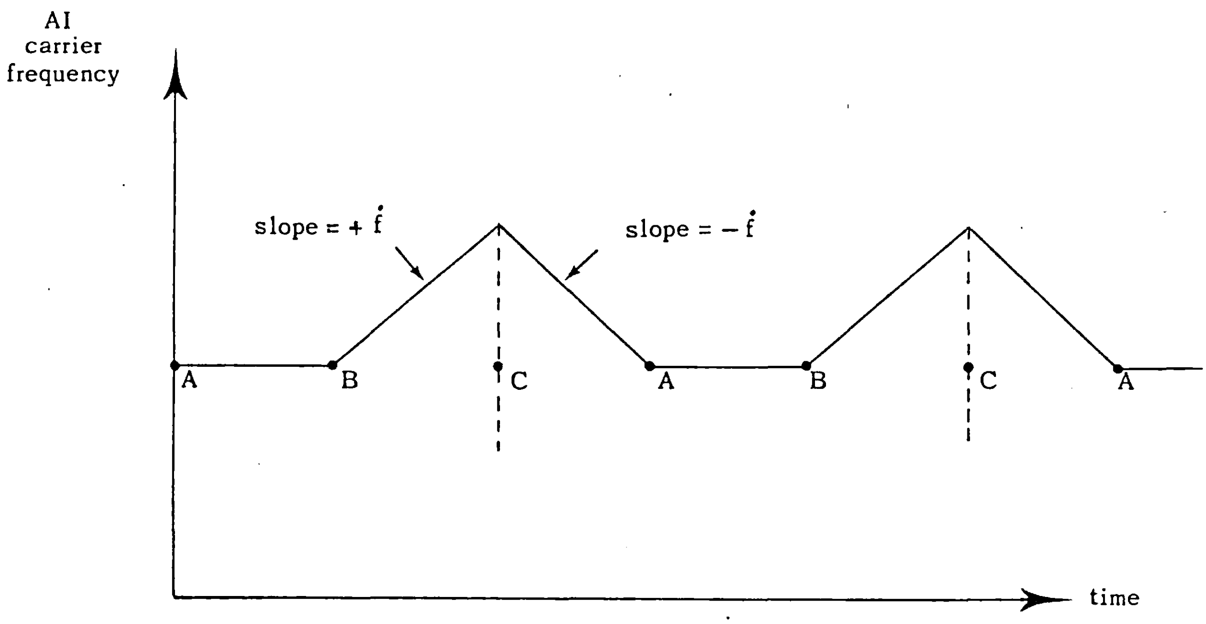


FIGURE 74.1. SHAPE OF AI CARRIER FREQUENCY LINEAR RANGING MODULATION – SUBROUTINE ZCLUTTERINIT

Notes:

(i) The times spent on each of the three components of the carrier frequency modulation cycle are assumed equal, ie

$$AB = BC = CA$$

(ii) If f_D denotes the doppler frequency shift of the signal returned from the target on AB, then the doppler frequency shifts on BC and CA are

$$f_D + \frac{2r\dot{f}}{c}$$

and

$$f_D - \frac{2r\dot{f}}{c}$$

respectively, where c is the velocity of light and r is the range from the fighter to the target.

75. SUBROUTINE ZEVALUATE(I)

This routine is called from subroutines ZCLUTTERCALC, ZGINT and ZITERATE, and from Function ZFUNC. If I=1 then for a given value of R2 = R^2 the corresponding X and Y^2 values are found, while if I=2 $\frac{\partial X}{\partial R^2}$ and $\frac{\partial Y^2}{\partial R^2}$ are also calculated.

The quantity HH, defined in (72.25) is first calculated:

$$HH = 1 + \frac{R^2 - 1}{2\left(1 + \frac{R_e}{h}\right)}$$

From (73.2), in terms of program variables this becomes:

$$HH = 1 + (R2-1).C \tag{75.1}$$

The value of X corresponding to this particular value of R^2 is then given by equation (72.24):

$$X = A1.R + A2.R2 + HH.B$$

(The variable A1,A2, B and C are all calculated at the beginning of subroutine ZCLUTTERCALC).

The corresponding value of $Y2 = Y^2$ is then obtained from equation (72.26):

$$X^2 + Y^2 + (HH)^2 = R^2$$

If I=2, then from (72.24):

$$\frac{\partial X}{\partial R^2} = \frac{\partial X}{\partial R} \cdot \frac{\partial R}{\partial R^2} = \frac{1}{2R}\left(A1 + 2.A2.R + B.\frac{\partial(HH)}{\partial R}\right)$$

ie

$$\frac{\partial X}{\partial R^2} \equiv DXDR2 = 0.5.\frac{A1}{R} + A2 + B.C \tag{75.2}$$

Similarly, $\partial Y^2/\partial R^2$ is also obtained from (72.24):

$$2X.\frac{\partial X}{\partial R^2} + \frac{\partial Y^2}{\partial R^2} + 2HH.\frac{\partial HH}{\partial R^2} = 1 \tag{75.3}$$

From (75.1):

$$\frac{\partial HH}{\partial R^2} = C \tag{75.4}$$

Hence in terms of program variables:

$$\frac{\partial Y^2}{\partial R^2} \equiv Y2DASH = 1 - 2(HH.C + X.DXDR2) \tag{75.5}$$

76. FUNCTION ZF

This function is called from subroutine ZGINT to evaluate the expression:

$$x^2 f(x) \tag{76.1}$$

where

$$x = R^2 \tag{76.2}$$

and

$$f(R) = \frac{G^2 \sigma_0 W}{R^3 \cos\phi} \tag{76.3}$$

Hence the function ZF is given by:

$$ZF = \frac{G^2 \sigma_0 W}{\cos\phi} \cdot R \tag{76.4}$$

where, in terms of program variables:

(i) W is given by equations (72.18), (72.25) and (73.2)

(ii) σ_0 is given by equations (72.20), (72.21) and (73.2)

(iii) The value of G^2 is obtained from subroutine GAINEFF

(iv) $\cos\phi$ is given by equations (72.19) and (73.2)

77. SUBROUTINE ZFUNC(XARG)

This routine is called from subroutine ZGINT as part of the calculation of the clutter integral. This is illustrated in equation (79.15) for the case of a singularity of $Y^2=0$ at $R^2=X_0$; in this case the control variable RVALUE is set equal to X_0 in subroutine ZCLUTTERCALC. If instead a singularity of $Y^2=0$ occurs at X_1 , the upper end of the integration range, then RVALUE= X_1, while if there is no singularity in the integrand of (79.15) in the range $[X_0, X_1]$, then RVALUE = -1.0 .

If RVALUE = -1.0 , then:

$$\text{ZFUNC} = \text{FUNC1} = x^2 f(x) \big/ \sqrt{g(x)} \tag{77.1}$$

while if RVALUE = X_0 or X_1 , then:

$$\text{ZFUNC} = \text{FUNC1-FUNC2} \tag{77.2}$$

$$= \frac{x^2 f(x)}{\sqrt{g(x)}} - \frac{CONST}{\sqrt{\left|\frac{1}{x} - \frac{1}{RVALUE}\right|}} \tag{77.3}$$

where

$$x = \frac{1}{XARG}$$

$$R^2 = x \tag{77.4}$$

and

$$g(x) = Y^2 \tag{77.5}$$

The expression CONST is calculated in subroutine ZGINT and is given by equations (79.19),(79.20) and (79.21).

The term FUNC2 is obtained immediately. The value of x in FUNC1 is obtained from (77.4) and subroutine ZEVALUATE is then called. This calculates the corresponding value of $g(x) = Y^2$, together with some intermediate variables necessary for the evaluation in Function ZF of $f(x)$:

$$ZF(x) = x^2 f(x)$$

FUNC1 is then given by:

$$FUNC1 = \frac{ZF(x)}{\sqrt{Y^2}} \tag{77.6}$$

78. SUBROUTINE ZGAINEFF(G2)

This routine is called from Function ZF to evaluate the radar gain $G2 = G^2$ in the direction of the clutter patch, given that the mainbeam of the radar is pointing at the target. The direction cosines of the target relative to the radar are (TX,TY,TZ). From Figure (72.1) and equations (72.2), (72.16) and (72.25), the coordinates of the clutter patch relative to the radar are

$$R(X, Y, -HH)$$

For a given value of R^2, with the corresponding value of Y^2 positive, the points corresponding to both $+Y$ and $-Y$ contribute to the clutter integral. The two values are expressed most simply by rewriting G^2 as

$$G^2(X, Y, Z) + G^2(X, -Y, Z)$$

(Note that unless the target is directly ahead of the fighter, G^2 will not be symmetric about $y=0$).

The angles A_1 and A_2 subtended at the radar by the target and clutter patches are first calculated:

$$\cos A_1 = (x.T_x + y.T_y - HH.T_z)/R$$
$$\cos A_2 = (x.T_x - y.T_y - HH.T_z)/R \qquad (78.1)$$

Function IANG then claculates IA1 and IA2, the angles nearest to A_1 and A_2 respectively at which gain patterns are specified. The radar gain, $G2$, is then given by:

G2=GAIN(IA1)+GAIN(IA2) (78.2)

(The GAIN array contains the square of the AI radar receiver gain pattern, G, and is calculated in subroutine ZCLUTTERINIT).

79. SUBROUTINE ZGINT(XØ,X1,DPOWER)

This routine is called from subroutine ZCLUTTERCALC to evaluate the expression

$$DPOWER = \int_{X_0}^{X_1} \frac{G^2 \sigma_0 W}{R^3 \cos\phi . |Y|} \, dR^2 \tag{79.1}$$

The limits of integration $X_0 = X\emptyset$ and $X_1 = X1$ are calculated in subroutine ZCLUTTERCALC:

$$1 \leq X_0 < X_1 \leq UPLIM = 1 + \frac{2R_e}{h} \tag{79.2}$$

There may be a simple zero of $Y = 0$ at X_0 or X_1, and there are no other such zeros in the range (X_0, X_1). The variable RVALUE is also set in the calling routine. This equals -1 if there is no singularity of the integrand, otherwise it equals the point, X_0 or X_1 , at which Y has a simple zero. The processing in this routine is expressed in terms of RVALUE and, without loss of generality, we may suppose that a singularity occurs at

$$RVALUE = X_0 \tag{79.3}$$

<u>Theory</u>

Two transformations of the integral in (79.1) are carried out. Firstly, the integrand varies more smoothly if the integration variable is taken as $\frac{1}{R^2}$, rather than R^2 . We write

$$x = R^2 \tag{79.4}$$

$$u = \frac{1}{x} = \frac{1}{R^2} \tag{79.5}$$

$$|Y| = \sqrt{Y^2} \tag{79.6}$$

and

$$DPOWER = \int_{X_0}^{X_1} \frac{f(x)}{\sqrt{g(x)}} \, dx \tag{79.7}$$

where $g(x)$ has a simple zero at $x = X_0$. Substituting (79.5) into (79.7):

$$\text{DPOWER} = \int_{U_1}^{U_0} \frac{f\left(\frac{1}{u}\right)}{u^2\sqrt{g\left(\frac{1}{u}\right)}}\, du \tag{79.8}$$

where

$$U_0 = \frac{1}{X_0}$$
$$U_1 = \frac{1}{X_1} \tag{79.9}$$

If we write

$$h(u) = g\left(\frac{1}{u}\right) \tag{79.10}$$

equation (79.8) becomes:

$$\text{DPOWER} = \int_{U_1}^{U_0} \frac{f\left(\frac{1}{u}\right)}{u^2\sqrt{h(u)}}\, du \tag{79.11}$$

where $h(u)$ has a simple zero at $u = U_0$.

To remove this singularity in the integrand, $h(u)$ is expanded as a Taylor series for points sufficiently near to U_0:

$$h(u) = h(U_0) + (u - U_0).\left.\frac{d(h(u))}{du}\right|_{u=U_0} + \ldots\ldots \tag{79.12}$$

Now,

$$\left.\frac{dh(u)}{du}\right|_{u=U_0} = \left.\frac{dg(x)}{dx}.\frac{dx}{du}\right|_{x=X_0} = -X_0^2.\left.\frac{dg}{dx}\right|_{X_0} \tag{79.13}$$

From Figure (73.3) $\frac{dg}{dx}$ is positive at X_0, so that $\frac{dh}{du}$ is negative at U_0. If h' denotes differentation with respect to u, then (79.11) may be written:

$$DPOWER = \int_{U_1}^{U_0} \left[\frac{f(\frac{1}{u})}{u^2 \sqrt{h(u)}} - \frac{f(\frac{1}{U_0})}{U_0^2 . \sqrt{|h'(U_0)| . (u-U_0)}} \right] du$$

$$+ \frac{f(\frac{1}{U_0}) . 2\sqrt{U_0 - U_1}}{U_0^2 . \sqrt{|h'(U_0)|}} \tag{79.14}$$

where the last two terms are equal and opposite. The integrand in (79.14) is now non-infinite at its limits of integration. Finally, re-writing it in terms of x gives the expression actually evaluated in subroutine GINT:

$$DPOWER = \int_{1/X_1}^{1/X_0} \left[\frac{x^2 f(x)}{\sqrt{g(x)}} - \frac{X_0^2 f(X_0)}{X_0 . \sqrt{g'(X_0)} . \sqrt{\frac{1}{x} - \frac{1}{X_0}}} \right] du + \left\{ \frac{X_0^2 f(X_0) . 2\sqrt{\frac{1}{X_0} - \frac{1}{X_1}}}{X_0 \sqrt{g'(X_0)}} \right\} \tag{79.15}$$

Processing

The integral is evaluated from equation (79.15). The integration range $\left[\frac{1}{X_1}, \frac{1}{X_0}\right]$ is split into NINTP equal sub-intervals:

$$[ARG(I), ARG(I+1)], \quad I=1,.....,NINTP \tag{79.16}$$

where

$$ARG(1) = 1/X_1$$
$$ARG(NINTP+1) = 1/X_0 \tag{79.17}$$

and NINTP is specified in the Block Data segment.

The value of the integrand at each of the points ARG(I) is calculated in Function ZFUNC; a standard Gaussian integration procedure is then applied to give the value of the integral, denoted by RESULT. The final answer is then:

$$DPOWER = RESULT + 2. CONST. \sqrt{\frac{1}{X_0} - \frac{1}{X_1}} \tag{79.18}$$

The constant term CONST is zero if RVALUE = -1.0 (no singularity at X_0 or X_1; otherwise it is given by:

$$CONST = \frac{ZF(RVALUE)}{\sqrt{|GRAD|} . RVALUE} \tag{79.19}$$

where the function ZF (x) is defined by:

$$ZF(x) = x^2 f(x) \tag{79.20}$$

(Subroutine ZEVALUATE must first be called in order for the function f to be evaluated). Finally, GRAD is calculated in subroutine ZCLUTTERCALC and is given by

$$\begin{aligned} \text{GRAD} &= g'(\text{RVALUE}) \qquad (79.21) \\ &= \left. \frac{\partial Y^2}{\partial R^2} \right|_{R^2=\text{RVALUE}} \end{aligned}$$

80. SUBROUTINE ZITERATE

This routine is called from subroutine ZCLUTTERCALC to find the value of R^2 corresponding to a root of $Y^2 = 0$. This enables the range of integration of the pulse doppler clutter integral, which includes $|Y|$ in the denominator, to be subdivided so that singularities are avoided.

Y^2 is expressed as a function of R^2 by equation (72.26):

$$X^2 + Y^2 + (HH)^2 = R^2$$

The initial estimate $(R^2)_0$ to the solution of $Y^2 = 0$ is calculated by binary chop in subroutine ZCLUTTERCALC. Subroutine ZEVALUATE calculates, for a given value of R^2, the corresponding values of Y^2 and $\partial Y^2 / \partial R^2$. A Newton-Raphson iterative approximation is carried out, currently for four cycles:

$$Y^2 = f(R^2) \qquad (80.1)$$

$$(R^2)_i = (R^2)_{i-1} - \frac{[f(R^2)]_{i-1}}{[f'(R^2)]_{i-1}} \qquad (i = 1, \ldots, 4) \qquad (80.2)$$

In terms of program variables, this becomes:

```
R2 = R2-Y2/Y2DASH
```

The number of roots found is increased by one:

```
NROOTS = NROOTS+1
```

The final approximation $(R^2)_4$ to this root of the equation $Y^2 = 0$ is stored in an array, as is the corresponding value of $\frac{\partial Y^2}{\partial R^2}$:

```
ROOTS(NROOTS)=R2
```
(80.3)

```
SLOPE(NROOTS)=Y2DASH
```
(80.4)

81. CALCULATION OF PULSE RADAR CLUTTER FOR AN AI - RADAR

Pulse radar clutter occurs when radar returns are received from points on the ground (or sea) at the same distance from the radar as the target of interest. More precisely, if r is the range from the radar to the target, f is the radar pulse repetition frequency and c is the velocity of light, clutter is received from all points on the ground distant r_c from the radar, where r_c is given by

$$r_c = r + \frac{nc}{2f} \qquad (81.1)$$

n is an integer, the 'order' of the clutter patch. n = 0 produces the normal clutter return, while the other possible cases represent the reception of returns from preceding or succeeding pulses. In practice, if (81.1) can be satisfied for $n < 0$, clutter returns will be from so much nearer than the target that detection is impossible, while for $n > 0$ the opposite applies, and clutter returns are negligible. In this section, therefore, it will be assumed that $r_c = r$ (though this actually depends on f being relatively low).

Any value of r_c obtained from (81.1) must satisfy two other conditions if clutter is to occur. Firstly the clutter patch must be nearer than the radar horizon:

$$r_c < \sqrt{2 R_e h} \qquad (81.2)$$

where R_e is the effective earth radius and h is the height of the radar above the ground.

Secondly the clutter patch must be at least as far away as the nearest ground surface:

$$r_c > h \qquad (81.3)$$

The clutter power dC received from an area element dA of ground surface is given by the appropriate form of the radar equation:

$$dC = \frac{P \lambda^2 \sigma_0 G^2 dA}{(4\pi)^3 . r^4} \qquad (81.4)$$

Where P = Transmitter peak power (including the effect of transmission and receiption losses)

λ = Wavelength of carrier

σ_0 = Ground radar scattering cross-section/unit area at the clutter patch

G = Radar gain in the direction of the clutter patch

To integrate (81.4) σ_o , G^2 and dA must be evaluated. The geometry of the situation is shown in Figure (81.1). θ_d is the depression angle from the radar to the clutter patch, θ_g is the grazing angle at the clutter patch and R_e is the effective earth radius .

Using the cosine rule, θ_d and θ_g may be obtained in terms of h,r, R_e (all known):

$$\sin \theta_g = \frac{h}{r} - \frac{r^2-h^2}{2rR_e} \qquad (81.5)$$

$$\sin \theta_d = \frac{r^2+h^2+2R_e h}{2r(R_e+h)} \qquad (81.6)$$

σ_o is only a function of θ_g, and may therefore be found immediately (it varies rapidly for small θ_g , which is why the earth's curvature must be taken into account). The theoretical expression used for σ_o is

$$\sigma_o = \sigma_{oo} (\sin \theta_g)^{\alpha} \qquad (81.7)$$

Here σ_{oo} is the value of σ_o at normal incidence. The values of the constants σ_{oo} and α , for each radar type, must be specified in the input data.

To find dA, we must look at the clutter patch in more detail - see Figure (81.2)

If t is the pulse duration, the distance l is given by

$$l = c.(\tfrac{1}{2}t) = \tfrac{1}{2}ct \qquad (81.8)$$

Now $\cos \theta_g = \frac{l}{dx}$, from which the element of length, dx, is given by

$$dx = \tfrac{1}{2}ct\sec \theta_g \qquad (81.9)$$

Note: This expression for the length of interception of the radar beam with the ground is not valid for very large θ_g. Then this distance is determined by the radar beamwidth $\Delta\theta$, and the pulse-length t is irrelevant. More precisely, equation (81.9) holds as long as

$$\tfrac{1}{2}ct.\sec \theta_g < r.\Delta\theta . \operatorname{cosec} \theta_g$$

ie as long as $\tfrac{1}{2}ct.\tan \theta_g < r.\Delta\theta$ (81.10)

Hence the total clutter area is a circular annulus of radius $r\cos\theta_d$ and width $\tfrac{1}{2}ct\sec\theta_g$ (see Figure 81.3). ϕ is the azimuthal bearing from the radar, with $\phi = 0$ in the direction of the target.

From Figure 81.3 we have:

$$dA = (r\cos\theta_d).(\tfrac{1}{2}ct\sec\theta_g).d\phi \tag{81.11}$$

If C denotes the total clutter power, then from equation (81.4):

$$C = \frac{P\lambda^2\,\sigma_0(\theta_g).ct.\cos\theta_d.\sec\theta_g}{(4\pi r)^3}\int_0^{\pi} G^2\,d\phi \tag{81.12}$$

Given the radar gain pattern, evaluation of the integral in (81.12) is straightforward.

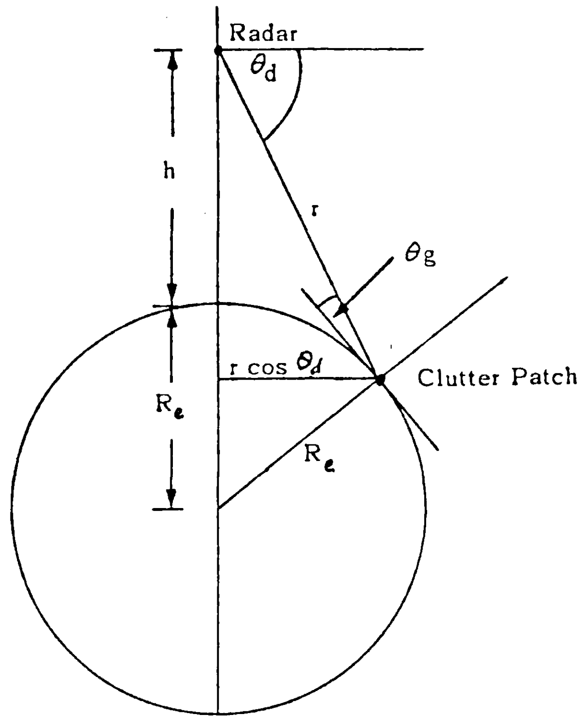


FIGURE 81.1. GEOMETRY OF THE RADAR AND THE PULSE CLUTTER PATCH

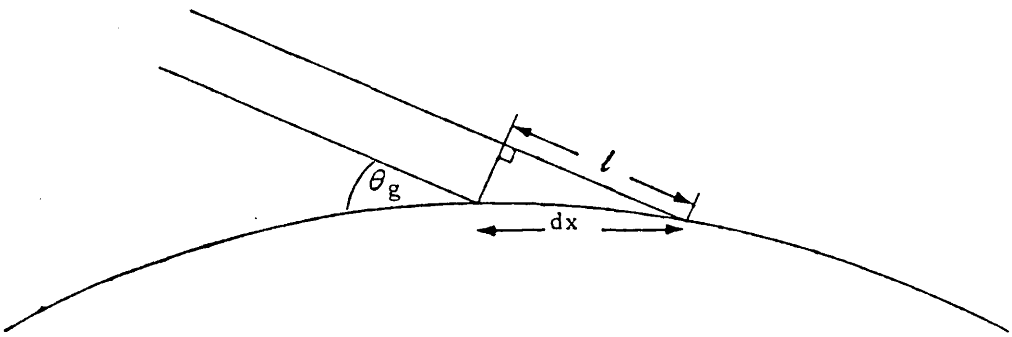


FIGURE 81.2. PULSE CLUTTER PATCH

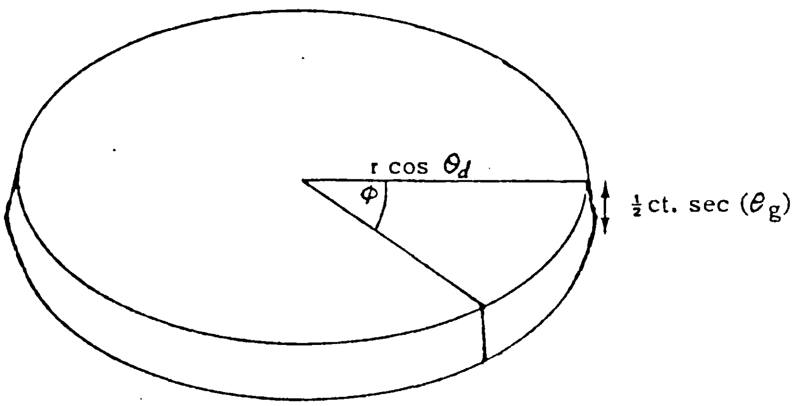


FIGURE 81.3. TOTAL PULSE CLUTTER AREA

82. SUBROUTINE ZPULSECL(RESULT)

This routine calculates the total pulse radar clutter experienced by a given fighter when its mainbeam illuminates a given target. It is not currently utilised in the fighter model . If the pulse clutter is denoted by RESULT, then from (81.12):

$$\text{RESULT} = \frac{P\lambda^2 \sigma_0\, c\tau . \cos\theta_d . \sec\theta_g}{(4\pi r)^3} \int_0^{\pi} G^2 \, d\phi$$

Subroutine ZPULSECLINIT is first called; this calculates those variables which are dependent on the current fighter and raid geometry. The depression angle θ_d from the radar to the clutter patch and the grazing angle θ_g at the clutter patch (Figure 81.1) may then be calculated from equations (81.5) and 81.6).

The integral in (81.12) is simplified by choosing a coordinate system for these calculations with the target in the (x,z)- plane (see Figure 82.1). Only one independent direction cosine, T_z ,is then needed to specify its orientation relative to the AI radar. The remaining direction cosine, T_x , is then given by:

$$T_x^2 + T_z^2 = 1$$

If (x,y,z) are the coordinates of a clutter patch with respect to the origin at the radar, then in terms of program variables:

$$z = -HH \qquad (82.1)$$

where from Figure 81.1 :

$$HH = r\sin\theta_d \qquad (82.2)$$

(r denotes the range from the radar to the target).

The range of integration $[0,\pi]$ is split up into NINTP equal subintervals, where NINTP is specified in the Block Data segment. The value of G^2 at each of the (NINTP +1) points ϕ(I) is then calculated in subroutine ZPULSEF, where

$$\phi(I) = \frac{(I-1).\pi}{NINTP} \qquad I=1,\ldots..,NINTP+1 \qquad (82.3)$$

Knowing these values a standard Gaussian integration procedure evaluates the integral

$$\int_0^{\pi} G^2 \, d\phi$$

The value of σ_0 is calculated from equation (81.7) and the solution to (81.12) is finally obtained:

$$\text{RESULT} = \frac{\text{PULSEPOWER}.(\sigma_0 . \cos\theta_d . \sec\theta_g)}{r^3} . \int_0^{\pi} G^2 \, d\phi \qquad (82.4)$$

PULSEPOWER is calculated in subroutine INPUT for each radar type and is given by:

$$\text{PULSEPOWER} = P_p . L_T . L_R . \frac{\lambda^3}{(4\pi)^3} . \frac{F}{f} \qquad (82.5)$$

where:

P_p = radar peak power

L_T = radar transmission loss

L_R = radar reception loss

λ = radar carrier wavelength

F = radar carrier frequency

f = radar pulse repetition frequency

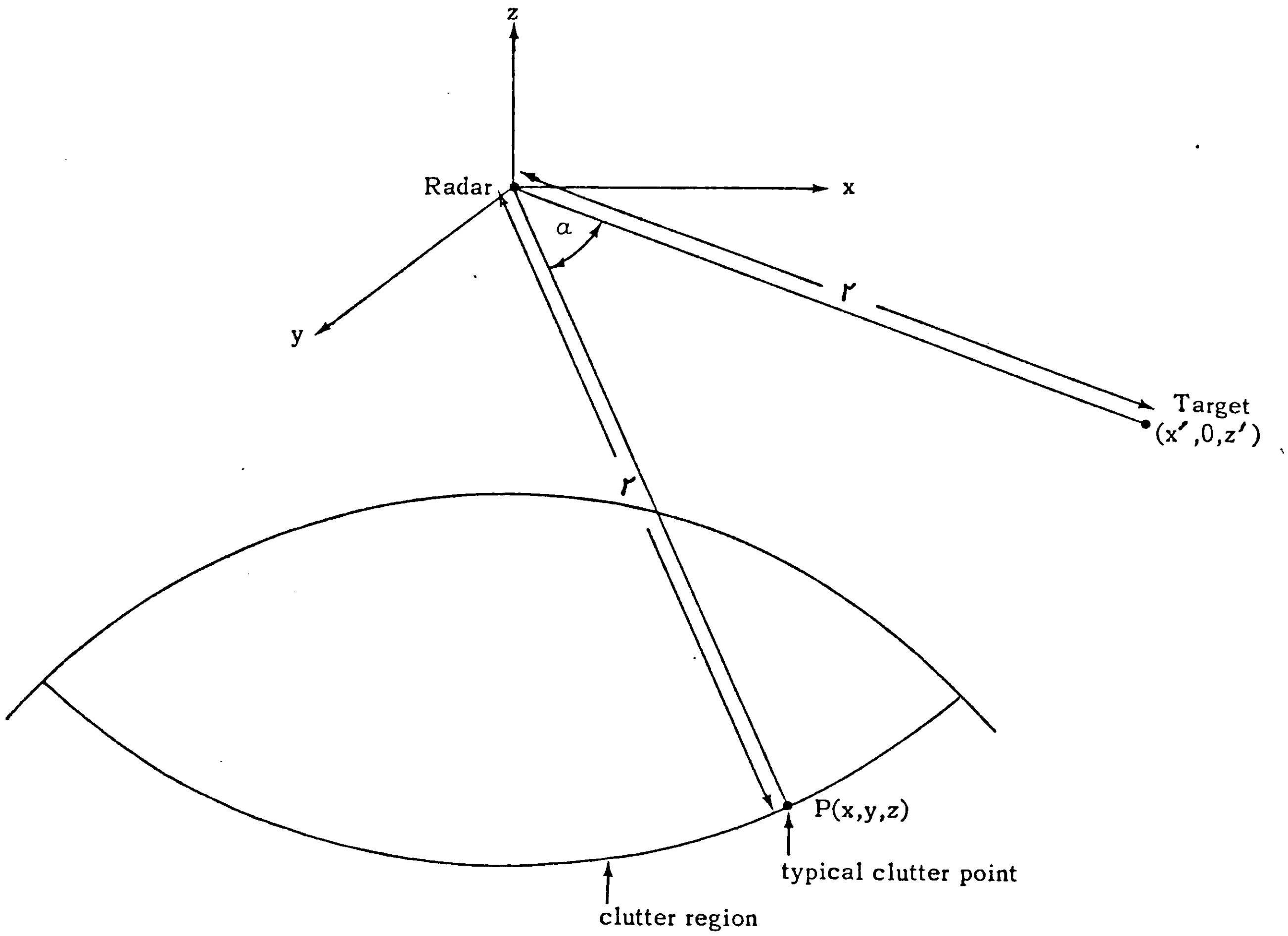


FIGURE 82.1. COORDINATE SYSTEM FOR PULSE RADAR CLUTTER CALCULATIONS – SUBROUTINE ZPULSECL

83. SUBROUTINE ZPULSECLINIT

This routine is called from subroutine ZPULSECL, which calculates the pulse clutter experienced by a given fighter when its mainbeam is illuminating a given target. It calculates the parameters which are dependent on the fighter type and the fighter and target geometry:

(i) r, the range from the fighter to the target

(ii) h, the fighter altitude, currently assumed to be equal to the raid altitude.

(iii) T_z, the z-direction cosine of the target with respect to the fighter. The coordinate system in subroutine ZPULSECL is such that the target is in the (x,z)-plane, so that only one independent direction cosine need be specified. Until the fighter's altitude is updated regularly in the model its attack is assumed to be at co-altitude with the raid, so that

$$T_z = 0.0$$

(iv) The ground scattering cross-section/unit area at the clutter patch, denoted by σ_o , is assumed given by equation (81.7) The values of the constants σ_{oo} = SIGØ and α = SIGEXP are specified in the input data for each radar type.

(v) The GAIN array contains G^2 ,where G is the AI radar 1-way gain pattern.

84. FUNCTION ZPULSEF(PHI)

This Function is called from subroutine ZPULSECL to evaluate G^2, the two-way radar gain at the clutter point P at an azimuth angle ϕ =PHI, when the mainbeam of the radar is illuminating the target (ϕ=0). This is illustrated in Figure 82.1, while Figure 84.1 presents a plan view of the geometry. The target, by definition of the coordinate system, is in the (x,z)-plane.

If $\hat{\underline{T}}$ denotes the unit position vector of the target relative to the radar, then:

$$\hat{\underline{T}} = (T_x, 0, T_z) \tag{84.1}$$

where T_x and T_z are known.

If $\underline{R}$ is the position vector of the clutter patch relative to the radar (at the origin of coordinates), then:

$$\underline{R} = (x, y, -HH) \tag{84.2}$$

HH is given by equation (82.2), while x is calculated as follows.

In Figure 84.1, RP represents the projection onto the horizontal plane of the line from the radar to the clutter patch; from Figure 81.1 this is given by:

$$|RP| = r.\cos\theta_d$$

Hence from Figure 84.1 the x-coordinate of the clutter patch is:

$$x = r\cos\theta_d . \cos\phi \tag{84.3}$$

From Figure 82.1 it is required to calculate the radar gain at an angle α, where α is the angle subtended at the radar by the target and clutter patches. This is given by:

$$\underline{R}.\hat{\underline{T}} = |\underline{R}|.|\hat{\underline{T}}|.\cos\alpha \tag{84.4}$$

ie

$$\cos\alpha = \frac{x.T_x - HH.T_z}{r} \tag{84.5}$$

The angle IA nearest to α at which radar gain patterns are specified is then calculated in Function IANG. The required 2-way radar gain, G^2, is then given by:

$$G^2 = GAIN(IA) \qquad (84.6)$$

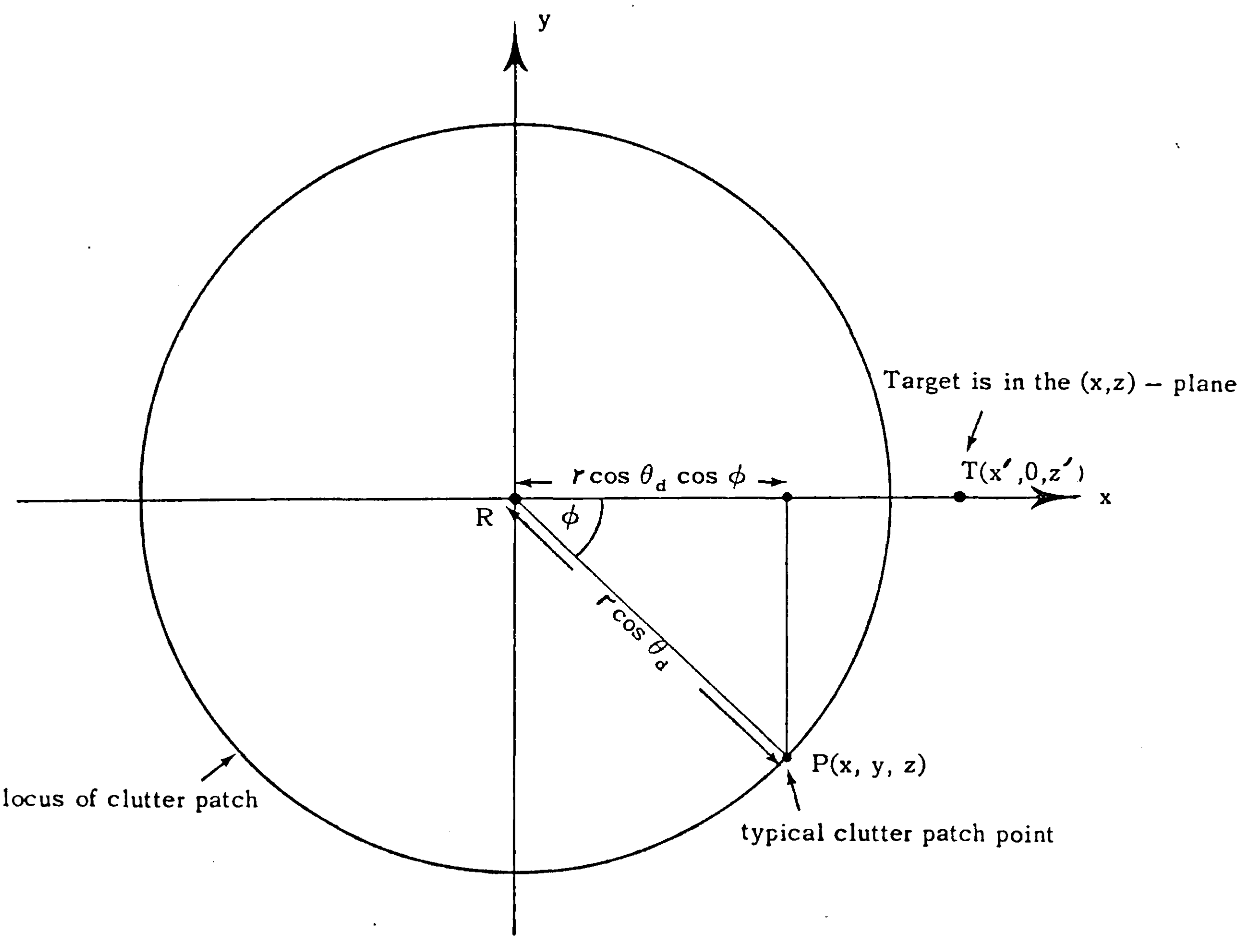


FIGURE 84.1. PLANE VIEW OF THE PULSE CLUTTER GEOMETRY – SUBROUTINE ZPULSEF

GLOSSARY OF VARIABLE NAMES

ALPORIG(IFTY)	INITIAL FIGHTER LEAD-PURSUIT ANGLE
ANGLP(IF)	CURRENT LEAD ANGLE OF FIGHTER IF'S LEAD-PURSUIT COURSE
ATTACKANGLE(IFTY)	THRESHOLD ANGLE BETWEEN FIGHTER AND RAID VELOCITIES WHICH DETERMINES IF AN ATTACK IS FORWARD/REAR HEMISPHERE
AVFREQ(IFTY)	AVERAGE CARRIER FREQUENCY FOR FIGHTER TYPE IFTY
BASEX(IBASE)	X-COORD OF EACH FIGHTER BASE
BASEY(IBASE)	Y-COORD OF EACH FIGHTER BASE
BEAMW(IRAD)	RADAR BEAMWIDTH OF RADAR TYPE (=FIGHTER TYPE) IRAD
BX1(IB)	X-COORD. OF ONE END OF OUT-OF-BOUNDS LINE IB
BX2(IB)	X-COORD. OF OTHER END OF OUT-OF-BOUNDS LINE IB
BY1(IB)	Y-COORD. OF ONE END OF OUT-OF-BOUNDS LINE IB
BY2(IB)	Y-COORD. OF OTHER END OF OUT-OF-BOUNDS LINE IB
CAPHT(IFTYPE)	CAP HEIGHT OF A FIGHTER TYPE
CBY4PI	VELOCITY OF LIGHT, C, DIVIDED BY 4PI
CDELAY(ICTYPE)	FIGHTER REACTION DELAY IN CONTROL MODE ICTYPE (=1,...,4) TO A RAID TRACK-CHANGE (SEE SUBROUTINE RESPACC)
CPATR(IFTY)	P.D. CLUTTER CONSTANT (SUBROUTINE ZCLUTTERCALC)
CVEL	SPEED OF LIGHT
DEGRAD	CONSTANT: CONVERTS DEGREES TO RADIANS
DELTFREQ(IFTY)	MINIMUM SEPARATION BETWEEN AI CARRIER FREQUENCIES (DETERMINISTIC MODE)
DETSNR(IFTY)	SIGNAL TO NOISE RATIO FOR INITIAL DETECTION OF TARGET
DS	DISTANCE OF A RAID FROM ITS TARGET AT MISSILE SPLASH (SUBROUTINES STATS AND WRITERAD)
DTLOOK	TIME INTERVAL BETWEEN RANGE CHECKS WHEN APPROACHING LAUNCH SUCCESS ZONE
DTR	TIME INTERVAL(SEC)BETWEEN RADAR ROUTINE CALLS
E	REQUIRED ACCURACY IN THE LOCALISATION OF SINGULARITIES IN A PULSE DOPPLER CLUTTER INTEGRAL
ECLIPS(IFTY)	ECLIPSING RADAR LOSS (AFFECTS TARGET AND CLUTTER ONLY)
EVENT(N)	EVENT TIME
EWT(IR,IEW)	TIME AT WHICH RAID IR CROSSES EW LINE IEW
EWX(10)	X-COORDS OF POINTS ON EW LINE
EWY(10)	Y-COORDS OF POINTS ON EW LINE
F	DOPPLER FREQUENCY SHIFT OF TARGET
FDIRERR(IFTY)	S.D. OF ERROR IN FIGHTER ESTIMATE OF TARGET HEADING
FDOT,FDOTPD(IFTY)	SLOPE OF LINEAR RANGING MODULATION OF CARRIER FREQUENCY
FFX(IF)	CURRENT X-COMP OF ACCELERATION OF FIGHTER IF
FFY(IF)	CURRENT Y-COMP OF ACCELERATION OF FIGHTER IF
FH(IF)	LATEST CALCULATED ALTITUDE OF FIGHTER IF
FLF(ICAP,IBASE,IFTY)	FUEL USED BY A FIGHTER IN CRUISING TO A CAP FROM A BASE
FLTOT	TOTAL NUMBER OF AIRCRAFT IN FLIGHT BEING CONSIDERED (SUBROUTINE RAIDKILL)
FNB(IFL,IR)	NUMBER OF BOMBERS IN FLIGHT IFL OF RAID IR
FNJ(IFL,IR)	NUMBER OF JAMMERS IN FLIGHT IFL OF RAID IR
FPOSERR(IFTY)	S.D OF ERROR IN FIGHTER ESTIMATE OF TARGET POSITION
FPPT(IPROF,IFTY,IPPT)	ACCELERATION BETWEEN PROFILE POINTS (IPPT,IPPT+1)
FRATE(IF)	CURRENT FUEL CONSUMPTION RATE OF FIGHTER IF
FRATIO	FIGHTER/ENEMY AIRCRAFT RATIO WHICH THE ALLOCATION PROCESS WILL ATTEMPT TO ACHIEVE WHEN THE STRENGTH OF THE RAID IS KNOWN
FREQ(IFTY)	NOMINAL CARRIER FREQUENCY OF FIGHTER TYPE IFTY
FRES(IFTY)	FUEL RESERVE NEEDED TO ATTEMPT AN INTERCEPTION
FSPEEDERR(IFTY)	S.D. OF ERROR IN FIGHTER ESTIMATE OF TARGET SPEED
FT(IF)	TIME AT WHICH POSITION OF FIGHTER IF LAST UPDATED
FUEL(IF)	CURRENT FUEL RESERVES OF FIGHTER IF
FUELI(IPROF,IFTY)	FUEL LEFT AFTER TAXI,RUN-UP AND TAKE-OFF
FUELRATE(IPROF,IFTY,I)	FUEL CONSUMPTION RATE AT PROFILE POINT I

FV(IF)	CURRENT VELOCITY OF FIGHTER IF
FVX(IF)	CURRENT X-COMP OF VELOCITY OF FIGHTER IF
FVY(IF)	CURRENT Y-COMP OF VELOCITY OF FIGHTER IF
FX(IF)	LATEST CALCULATED X-COORD OF FIGHTER IF
FY(IF)	LATEST CALCULATED Y-COORD OF FIGHTER IF
GAIN	DUMMY ARRAY IN SUBROUTINE GAINREAD INTO WHICH POLAR DIAGRAMS ARE READ (ALSO 2-WAY AI RADAR GAIN PATTERN, IN CLUTTER CALCULATIONS)
GANGLE	ARRAY OF POINTS AT WHICH A POLAR DIAGRAM IS SPECIFIED
GDIRERR	S.D. OF ERROR IN GROUND CONTROL REPORTING OF TARGET HEADING
GFAC(IFTY)	SWERLING'S G-FACTOR FOR RADAR CALCULATIONS
GJR(IANG,IJTYPE)	1-WAY JAMMER RECEIVER GAIN PATTERN
GJT(IANG,IJTYPE)	1-WAY JAMMER TRANSMITTER GAIN PATTERN
GR1(IANG,IRAD)	1-WAY RADAR GAIN PATTERN OF RADAR IRAD(ABS-NOT DB)
GRMAX(IFTY)	MAX 1-WAY RADAR GAIN OF RADAR IFTY(ABS-NOT DB)
GROUNDELAY	DELAY BEFORE WARNING-LINE DATA FROM GROUND CONTROL IS MADE AVAILABLE
GROUNDELAY1	DELAY BEFORE TRACK-CHANGE DATA FROM GROUND CONTROL IS MADE AVAILABLE
GROUNDERROR	S.D. OF ERROR IN G.C. REPORTING OF TARGET POSITION
GSPEEDERR	S.D. OF ERROR IN G.C. REPORTING OF TARGET SPEED
GVALUE	ARRAY OF VALUES OF A POLAR DIAGRAM CORRESPONDING TO THE ANGLES SPECIFIED IN THE GANGLE ARRAY
H	FIGHTER ALTITUDE (CLUTTER CALCULATIONS)
HERROR(ICTYPE)	HEADING ERROR APPLIED TO INTERCEPTION COURSES OF FIGHTER CONTROL MODE ICTYPE(=1,..,4);SEE SUBROUTINE RESPACC
HOJEND(IFTY)	JAMMING TO NOISE RATIO FOR TERMINATION OF HOME-ON-JAM AND ANGLE-ON-JAM MODES
HOJENDS(IFTY)	SIGNAL TO JAMMING RATIO FOR TERMINATION OF HOME-ON-JAM AND ANGLE-ON-JAM MODES
HOJSNR(IFTY)	JAMMING TO NOISE RATIO FOR HOME-ON-JAM INITIATION
HOJSTARTS(IFTY)	SIGNAL TO JAMMING RATIO FOR HOME-ON-JAM INITIATION
HR(IT,IR)	ALTITUDE AT START OF TRACK IT OF RAID IR
I12OR16	=1/0 IF EVENT TYPE 12/16 IS THE MEASURE OF FIGHTER SUCCESS
IANG	SEE FUNCTION IANG
IATTACK(IR,I)	ARRAY OF SERIALS OF ALL FIGHTERS ATTACKING RAID IR (1.LE.I.LE.21)
IB	RADAR BAND SERIAL (=1 OR 2)
IBASE	SERIAL OF A FIGHTER BASE (.LE.3)
IBW	JAMMER BAND BANDWIDTH (MHZ)
ICALL	IDENTIFIES THE ROUTINE CALLING SUBROUTINE INTCALCONTROL TO CALCULATE AN INTERCEPTION COURSE (=1,...,7)
ICAP	SERIAL OF A CAP
ICC	=0 IF CLOSE CONTROL NOT POSSIBLE =1 IF CLOSE CONTROL IS POSSIBLE
ICOLLIDE(IF)	=1 IF FIGHTER IF HAS ACHIEVED BURNTHROUGH & ADOPTED AN INTERCEPTION COURSE,AND ALSO EVENT TYPE 12 IS THE MEASURE OF FIGHTER SUCCESS; =-1 OTHERWISE
ICTYPE	FIGHTER CONTROL MODE; SEE SUBROUTINE RESPACC
IDL(IFTY)	=1/0 IF FIGHTER TYPE IFTY HAS/DOES NOT HAVE A DATA LINK
IDOWN(IF)	ARRAY OF DOWNWARD POINTERS FOR FIGHTER SERIALS
IDUMC(IBASE)	ENSURES THAT INFORMATION THAT BASE OUT OF FIGHTERS IS OUTPUT ONCE ONLY
IDUMF	=1/0 IF FIGHTER POSITIONS ARE/ARE NOT PRINTED
IDUMR	=1/0 IF RAID POSITIONS ARE/ARE NOT PRINTED
IEV	TYPE OF EVENT TO BE NEXT EXECUTED IN SUBROUTINE EVENTS
IEV4(IR)	=0/1 IF RAID TRACK INFORMATION NOT/IS AVAILABLE
IEV5(IR)	=0/1 IF RAID SIZE INFORMATION NOT/IS AVAILABLE
IEVENT(1,N)	POINTER TO NEXT EVENT OR FREE CELL
IEVENT(2,N)	EVENT TYPE NUMBER
IEVENT(3,N)	ENTITY TO WHICH THE EVENT OCCURS
IEW	EARLY WARNING LINE SERIAL (=1,2 OR 3)
IF	SERIAL OF FIGHTER (.LE.21)
IF1	SERIAL OF 1ST ACCESSED FIGHTER; SEE SUBROUTINE SETIF

IF2	SEE SUBROUTINE SETIF
IFIRE(I)	=1 IF A RAID IS WITHIN THE LAUNCH ENVELOPES OF MISSILE TYPE I; =0 OR -1 OTHERWISE (SUBROUTINE LAUNCHTEST)
IFIREPT(IR)	WEAPON RELEASE POINT OF RAID IR
IFL	SERIAL OF ENEMY FLIGHT
IFLIGHT(IF)	SERIAL OF FLIGHT BEING ATTCKED BY FIGHTER IF (ONLY IF IT IS WITHIN THE REATTACK SEQUENCE; -1 OTHERWISE)
IFLNEXTO(I)	ARRAY OF POSSIBLE NEXT TARGET FLIGHTS (REATTACK ROUTINE)
IFM(MODE)	SERIAL OF FIRST FIGHTER AVAILABLE IN CONTROL MODE MODEC(IF), IF MODE=MODEC(IF)=1,2,3 OR 4
IFM(5)	FIRST AVAILABLE FIGHTER SERIAL FOR ALLOCATION
IFMIN(I)	ARRAY OF FIGHTERS WHICH MAY BE ALLOCATED TO INTERCEPT RAID IR (SUBROUTINE FFIND)
IFREP(IBASE)	FIGHTER TYPE FROM BASE IBASE (SUBROUTINE FCHOOSE)
IFREQ1(IJ,JJTYPE,IB)	MINIMUM FREQUENCY DETECTED BY JAMMER (IJ,IJTYPE) IN BAND IB
IFREQ2(IJ,IJTYPE,IB)	MAXIMUM FREQUENCY DETECTED BY JAMMER (IJ,IJTYPE) IN BAND IB
IFREQF(IF)	RADAR FREQUENCY OF FIGHTER IF(MHZ)
IFREQMIN	LOWEST RADAR FREQUENCY CONSIDERED
IFREQMAX	HIGHEST RADAR FREQUENCY CONSIDERED
IFTY	FIGHTER TYPE (.LE.3)
IFTYPE	FIGHTER TYPE (.LE.3)
IGVSF	INPUT VARIABLE,TO GOVERN RESPONSE OF A FIGHTER ON A CC OR DL COURSE;IGVSF=1 IF CONTINUE ON ORIGINAL COURSE; IGVSF=0 IF ADOPT A L.P. COURSE
IHEM(IF)	=1/2 IF FIGHTER IF HAS LAUNCHED A MISSILE IN A FORWARD/REAR HEMISPHERE ATTACK
IJ	JAMMER SERIAL (.LE.20)
IJTYPE	JAMMER TYPE,CURRENTLY: 1=SPECIALIST 2=NON-SPECIALIST(E.G. POD) (SOJ'S ARE NON-RESPONSIVE AND ARE PROCESSED SEPARATELY)
ILREP(IBASE)	READINESS LEVEL AT BASE IBASE (SUBROUTINE FCHOOSE)
IMPOSS	=0/1 IF INTERCEPTION IS/IS NOT POSSIBLE
INIT	NEAREST PROFILE POINT WITH A CORRESPONDING SPEED .LE. CURRENT FIGHTER SPEED
INITCAP(IR)	SERIAL OF CAP MOUNTED IN RESPONSE TO INITIAL DETECTION OF RAID IR
INOCHANGE	=0/1 IF CHANGE/NOCHANGE IN TRACK PARAMETERS
INTPHI	=IANG(PHI(IF,IJ))=ANGLE NEAREST TO PHI AT WHICH RADAR AND JAMMER GAIN PATTERNS ARE SPECIFIED
IPPT	SERIAL OF A PROFILE POINT
IPROF	SERIAL OF A FIGHTER PROFILE
IR	RAID SERIAL (.LE.20)
IREP	SERIAL OF CURRENT REPLICATION
IRLETH(IF)	INFRA-RED MISSILE LETHALITY
ISEED	RANDOM NUMBER SEED
IT	SERIAL OF ENEMY RAID TRACK:WORK VARIABLE
ITARGET(IF)	SERIAL OF FIGHTER IF'S TARGET RAID
ITR(IR)	SERIAL OF POINT @ END OF CURRENT TRACK OF RAID IR
ITURN(IF)	=1 IF FIGHTER IS TURNING AFTER AN ATTACK; =0 OTHERWISE
ITWS(IFTY)	=1/0 IF TRACK WHILE SCAN IS/IS NOT POSSIBLE
IUP(IF)	ARRAY OF UPWARD POINTERS FOR FIGHTER SERIALS
JC	MEASURE OF CURRENT ECM CONDITIONS (EVENT TYPE 16): =1 (CLEAR CONDITIONS);=2 (HOME-ON-JAM MODE); =3 (ANGLE-ON-JAM MODE)
JDET(IF,IJ,IJTYPE)	INDICATES WAY IN WHICH RADAR OF IF DETECTED BY JAMMER (IJ,IJTYPE) 0=UNDETECTED 1=CONTINUOUSLY DETECTED 2=DETECTED DURING DWELL-TIME ONLY
JMPX(IJTYPE)	=1/0 IF JAMMER ABLE/UNABLE TO DETECT AND RESPOND TO A SCANNING RADAR DURING PAINT ONLY
JPPT(IF)	LAST PROFILE POINT REACHED BY FIGHTER IF
KBASE(IF)	BASE OF FIGHTER IF
KCAP(IF)	CAP TO WHICH FIGHTER IF IS ASSIGNED
KDET	WORK VARIABLE HOLDING ONE ELEMENT OF JDET

KFTYPE(IF)	TYPE OF FIGHTER IF
KPROF(IF)	CURRENT FLIGHT PROFILE OF FIGHTER IF
LOCEVR(IR)	SERIAL OF NEXT EVENT DUE TO OCCUR TO RAID IR(=0 IF RAID IS ANNIHILATED OR HAS REACHED THE END OF ITS MISSION)
LOCEVF1(IF)	HOLDS SERIAL OF EVENT 7 OR 8
LOCEVF2(IF)	HOLDS SERIAL OF EVENT 9,10,11,12 OR 14
LOCEVF3(IF)	HOLDS SERIAL OF EVENT 15,16,17 OR 18
MAXFREQ(IFTY)	MAXIMUM RADAR CARRIER FREQUENCY FIGHTER TYPE IFTY
MINFREQ(IFTY)	MINIMUM RADAR CARRIER FREQUENCY FIGHTER TYPE IFTY
MISSMODE(MTYPE)	=1/2 IF MISSILE TYPE MTYPE IS SEMI-ACTIVE/INFRA-RED
MODEC(IF)	CONTROL MODE OF FIGHTER IF =1 IF FIGHTER ON OR CRUISING TO CAP =2 IF FIGHTER UNDER BROADCAST CONTROL =3 IF FIGHTER UNDER CLOSE CONTROL =4 IF FIGHTER UNDER AUTONOMOUS CONTROL
MODER(IF)	DETAILED CONTROL MODE OF FIGHTER IF =1 FIGHTER ON OR CRUISING TO CAP =2 FULL INFORMATION INTERCEPTION, RANGE VIA NON-TRACK -WHILE SCAN RADAR =3 UNJAMMED FIGHTER RADAR LOCKED ON TO TARGET DURING MISSILE FLIGHT TIME (RADAR MISSILE) =4 ATTACK WITH RADAR MISSILE, HOME-ON-JAM MODE (RANGE KNOWN) =5 INITIAL INTERCEPTION COURSE UNDER BROADCAST CONTROL =6 LEAD PURSUIT COURSE, TARGET DETECTED (RANGE UNKNOWN) =7 RADAR LOCKED ON TO JAMMING TARGET IN ANGLE-ON-JAM MODE, (RANGE UNKNOWN) =8 INTERCEPTION USING TRACK-WHILE-SCAN RADAR =9 INTERCEPTION IN HOME-ON-JAM MODE (RANGE KNOWN) =10 RADAR LOCKED ON TO JAMMING TARGET IN HOME-ON-JAM MODE =11 FIGHTER ON A CLOSE-CONTROLLED INTERCEPTION =12 FIGHTER ON A DATA LINK CONTROLLED INTERCEPTION =13 CLOSE CONTROL OR DL CONTROL,WITH DETECTION BUT NO RANGE;SAME AS MODER(IF)=6,BUT DO NOT WANT FIGHTER TO FOLLOW A LEAD PURSUIT COURSE
MODERUN	=0/1 FOR DETERMINISTIC/STOCHASTIC RUN
MPK(MTYPE,JC,IHEM)	PROBABILITY OF KILL OF MISSILE TYPE MTYPE IN ECM CONDITIONS JC AND ATTACK IN HEMISPHERE IHEM
MPK1	PROBABILITY OF KILL (PERCENTAGE)
MPRIOR(1,IFTY)	PREFERRED MISSILE TYPE IN FORWARD HEMISPHERE ATTACK
MPRIOR(2,IFTY)	PREFERRED MISSILE TYPE IN REAR HEMISPHERE ATTACK
MTYPE	MISSILE TYPE (=1/2 IF RADAR MISSILE/INFRA-RED)
MTYPEF(IF)	TYPE OF FIGHTER IF'S MISSILE CURRENTLY IN FLIGHT
N	SEE NCELLS (REF. TO ARRAYS IEVENT AND EVENT)
NACCESS(IBASE,ICAP)	PT.OF ACCESS OF A BASE TO A CAP
NATTACK(IR)	NUMBER OF FIGHTERS ATTACKING RAID IR
NBO(IFL,IR)	INITIAL NO. OF BOMBERS/SSJ IN FLIGHT IFL OF RAID IR
NB(IFL,IR)	NO. OF BOMBERS/SSJ IN FLIGHT IFL OF RAID IR
NBOUNDS	NUMBER OF OUT OF BOUNDS LINES LIMITING FIGHTER FLIGHT
NCAPS	NO. OF FIGHTER CAPS(.LE.3)
NCELLS	NO. OF CELLS AVAILABLE,I.E. THE SAME AS THE SECOND DIMENSION OF ARRAYS EVENT AND IEVENT;CURRENTLY NCELLS=200
NEWMODE(IF)	NEW VALUE OF MODEC(IF) IF THIS IS CHANGED IN SUBROUTINE RADEVENTS
NEWP	NO. OF POINTS SPECIFYING AN EARLY WARNING SEGMENT
NEXTCAPPT(IF)	NEXT CAP POINT FOR FIGHTER IF
NFO	NO.OF FIGHTERS SCRAMBLED TO A CAP AT FIRST DETECTION OF A RAID
NF1	INITIAL NO. OF FIGHTERS COMMITTED TO A RAID OF UNKNOWN STRENGTH WHEN THE RAID TRACK IS KNOWN
NF	NO. OF ACTIVE FIGHTERS IN MODEL
NFBASES	NO. OF FIGHTER BASES(.LE.3)
NFCUM(IBASE)	NO. OF FIGHTERS ON BASE IBASE

NFIGHTCUM	TOTAL NO. OF FIGHTERS WHICH HAVE TAKEN-OFF
NFIGHTER(IBASE,IFTYPE)	CURRENT NO.OF FIGHTERS OF EACH TYPE ON EACH BASE
NFIGHTERO(IBASE,IFTYPE)	INITIAL NO.OF FIGHTERS OF EACH TYPE ON EACH BASE
NFLO(IR)	INITIAL NO. OF FLIGHTS IN RAID IR
NFL(IR)	CURRENT NO. OF FLIGHTS IN RAID IR
NFTYPES	NO.OF FIGHTER TYPES(.LE.3)
NGC	CURRENT NUMBER OF GROUND CONTROLLERS AVAILABLE
NGCO	INITIAL NUMBER OF GROUND CONTROLLERS AVAILABLE
NINTP	NO. OF POINTS AT WHICH THE INTEGRAND IS SPECIFIED IN THE GAUSSIAN INTEGRATION FORMULA USED FOR CLUTTER CALCS.
NJB	NO. OF JAMMER BANDS (.LE.2)
NJO(IFL,IR)	INITIAL NO. SPECIALIST JAMMERS IN FLIGHT IFL OF RAID IR
NJ(IFL,IR)	CURRENT NO. SPECIALIST JAMMERS IN FLIGHT IFL OF RAID IR
NJTYPES	NO. OF DIFFERENT JAMMER TYPES(EXCLUDING SOJ) (.LE.2)
NLEVS	NO. OF READINESS LEVELS (.LE.4)
NMISS(1,IF)	NUMBER OF MISSILES OF TYPE 1 - RADAR CONTROLLED
NMISS(2,IF)	2 - INFRA-RED HELD BY FIGHTER IF
NMISSO	INITIAL NUMBER OF MISSILES HELD BY A FIGHTER
NMTYPES	NUMBER OF AIR-TO-AIR MISSILE TYPES (CURRENTLY =2)
NOEVPRINT(50)	LIST OF EVENT TYPES NOT TO BE LISTED IN DIAGNOSTIC PRINT-OUT
NPOINTS(ICAP)	NO.OF POINTS DEFINING EACH CAP(.LE.6)
NPROFILES	NO. OF FIGHTER PROFILES(.LE.8)
NPTS(IPROF,IFTY)	NO. OF POINTS SPECIFYING EACH PROFILE FOR EACH FIGHTER TYPE (.LE.7)
NQ(IBASE)	NO. OF FIGHTERS QUEUEING FOR TAKE-OFF AT BASE IBASE
NRO	INITIAL NO. OF ENEMY RAIDS
NR	CURRENT NO. OF ENEMY RAIDS
NRAD	NUMBER OF DIFFERENT RADAR TYPES
NREP	NO. OF MONTE CARLO REPLICATIONS
NRLEV(IBASE,IFTY,ILEV)	CURRENT NO. OF FIGHTER TYPE IFTY AT BASE IBASE AND READINESS LEVEL ILEV
NRLEVO(IBASE,IFTY,ILEV)	INITIAL NO. OF FIGHTER TYPE IFTY AT BASE IABASE AND READINESS ILEV
NSEARCH	NO. OF POINTS AT WHICH AN INTEGRAND IS CHECKED FOR SINGULARITIES IN THE CALCULATION OF A P.D. CLUTTER INTEGRAL
NSOJ	NUMBER OF STAND-OFF JAMMERS
NT(IR)	NO.OF TRACK-CHANGE POINTS IN TRACK OF RAID IR
NXTEVT	POINTER TO THE EARLIEST EVENT,THE NEXT TO BE PROCESSED
NXTFRE	POINTER TO THE 1ST.CELL OF THE LINKED LIST OF FREE CELLS
PC	RADAR CLUTTER
PDETJ(IJTYPE)	MINIMUM PEAK RADAR POWER DETECTABLE BY JAMMER IJTYPE (WATTS)
PE,PERR(IF)	RANDOM VARIABLE FOR RAID POSITIONAL ERROR CALCULATIONS
PHI(IF,IJ)	ANGLE BETWEEN VELOCITY VECTOR OF RAID IJ & ITS POSITION VECTOR RELATIVE TO FIGHTER IF
PI	CONSTANT,PI
PIX4	CONSTANT,4 MULTIPLIED BY PI
PJO(IJTYPE,IB)	POWER ALLOCATED TO BAND IB BY JAMMER TYPE IJTYPE
PJ(IJ,IJTYPE,IB)	POWER(WATTS) EMITTED BY JAMMER(IJ,IJTYPE)IN BAND IB
PJATR(IFTY)	JAMMER POWER AT A RADAR(SEE SUBROUTINE JPOWER)
PJMIN(IFTY)	THRESHOLD FOR DETECTION OF JAMMING(SUBROUTINE RADEVENTS)
PN	INTERNAL RADAR NOISE
POL	POLARIZATION LOSS FOR JAMMERS
POWER	TOTAL JAMMING POWER SUFFERED BY A GIVEN FIGHTER WHEN IT IS ILLUMINATING A GIVEN RAID
POWERJ	=POWER;SEE SUBROUTINE RADAR
PRF(IFTY)	RADAR P.R.F. (HZ)
PSOJ(IB,ISOJ)	TRANSMITTER POWER FOR STAND-OFF JAMMERS
PT	TARGET SIGNAL STRENGTH
PULSEPOWER(IFTY)	RADAR CONSTANT (DEFINED IN SUBROUTINE INPUT)
PX	X COORD. OF INTERCEPTION POINT RELATIVE TO FIGHTER'S INITIAL POSITION

PX1(I)	ARRAY OF X-COORDS. OF INTERCEPTION POINTS (SUBROUTINE FFIND)
PY	Y COORD. OF INTERCEPTION POINT RELATIVE TO FIGHTER'S INITIAL POSITION
PY1(I)	ARRAY OF Y-COORDS. OF INTERCEPTION POINTS (SUBROUTINE FFIND)
QE,QERR(IF)	RANDOM VARIABLE FOR RAID POSITIONAL ERROR CALCULATIONS
R1	RANGE TO RAID
R2J	SQUARED RANGE TO A JAMMER
R2R	SQUARED RANGE TO A RAID
RADMPOWER(IFTY)	MEAN RADAR POWER
RADPPOWER(IFTY)	RADAR PEAK POWER
RADPRF(IFTY)	RADAR PULSE REPETITION FREQUENCY
RAIDVX	ESTIMATED X COMPONENT OF RAID VELOCITY IN INTERCEPTION COURSE CALCULATION
RAIDVY	ESTIMATED Y COMPONENT OF RAID VELOCITY IN INTERCEPTION COURSE CALCULATION
RANGEMAX	RANGE ABOVE WHICH RADAR-RAID-JAMMER INTERACTIONS IGNORED
RBW(IRAD)	RECEIVER BANDWIDTH OF RADAR IRAD(MHZ)
RDIRERR(IR)	S.D. OF ERROR IN ESTIMATION OF HEADING OF RAID IR; SEE SUBROUTINES INTCALCONTROL AND UPDATEERROR
RE,RERR(IF)	RANDOM VARIABLE FOR RAID SPEED ERROR CALCULATIONS
RE	EFFECTIVE EARTH RADIUS (CLUTTER CALCULATIONS)
RELX	ESTIMATED X COORD OF RAID RELATIVE TO FIGHTER IN INTERCEPTION COURSE CALCULATION
RELXJ	X-COORD. OF RAID IJ RELATIVE TO FIGHTER IF
RELXR	X-COORD. OF RAID IR RELATIVE TO FIGHTER IF
RELY	ESTIMATED Y COORD OF RAID RELATIVE TO FIGHTER IN INTERCEPTION COURSE CALCULATION
RELYJ	Y-COORD. OF RAID IJ RELATIVE TO FIGHTER IF
RELYR	Y-COORD. OF RAID IR RELATIVE TO FIGHTER IF
REP	NO. OF REPLICATIONS (SUBROUTINE OUTPUT)
REPMIN1	(REP-1)
RESFAC	RESOLUTION FACTOR FOR RADAR EQUATION
RH(IR)	LATEST CALCULATED ALTITUDE FOR RAID IR
RJ	DISTANCE OF RAID IJ FROM FIGHTER IF
RL(IR)	CURRENT LENGTH OF RAID IR
RLO(IR)	INITIAL RAID LENGTH
RLEV(ILEV)	TIME BEFORE TAKE-OFF IN READINESS LEVEL ILEV
RLOSSR(IFTY)	RADAR RECEPTION LOSS
RLOSST(IFTY)	RADAR TRANSMISSION LOSS
RMAX(IANG,MTYPE)	MAX MISSILE LAUNCH SUCCESS RANGE
RMIN(IANG,MTYPE)	MIN MISSILE LAUNCH SUCCESS RANGE
RNOISE(IFTY)	RADAR RECEIVER NOISE POWER
RPATJ(IFTY)	RADAR POWER AT A JAMMER (SEE SUBROUTINE RADAR)
RPATR(IFTY)	RADAR POWER RETURNED TO THE RADAR(SUBROUTINE RADAR)
RPOSERR(IR)	S.D. OF ERROR IN ESTIMATION OF POSITION OF RAID IR
RR	DISTANCE OF RAID IR FROM FIGHTER IF
RSCANR(IFTY)	RADAR SCANNING RATE
RSNR(IFTY)	SIGNAL TO NOISE RATIO FOR SATISFACTORY RANGE DETERMINATION
RSPEEDERR(IR)	S.D. OF ERROR IN ESTIMATION OF SPEED OF RAID IR
RT(IR)	TIME AT WHICH POSITION AND VELOCITY OF RAID IR LAST UPDATED
RV(IR)	CURRENT VELOCITY OF RAID IR
RVT	ESTIMATED RAID SPEED IN SUBROUTINES INTCALCONTROL AND COLLISION
RVX(IR)	CURRENT X-COMP OF VEL OF RAID IR
RVY(IR)	CURRENT Y-COMP OF VEL OF RAID IR
RW(IR)	WIDTH OF RAID IR
RX(IR)	LATEST CALCULATED X-COORD OF RAID IR
RX1	X-COORD. OF A RAID AT A MISSILE SPLASH-POINT (SUBROUTINES STATS AND WRITERAD)
RY(IR)	LATEST CALCULATED Y-COORD OF RAID IR

RY1	Y-COORD. OF A RAID AT A MISSILE SPLASH-POINT (SUBROUTINES STATS AND WRITERAD)
SAFEFAC	FACTOR BY WHICH ACTUAL LAUNCH RANGE REDUCED FROM MAX MISSILE LAUNCH SUCCESS RANGE
SCANW(IFTY)	RADAR AZIMUTH SCAN HALF-WIDTH
SE,SERR(IF)	RANDOM VARIABLE FOR RAID HEADING ERROR CALCULATIONS
SIGEXP,SIGEXPFN(IFTY)	EXP OF SIN(GRAZING ANGLE) IN GROUND CROSS-SECTION FORMULA (SEE CLUTTER CALCULATIONS)
SIGO,SIGOFN(IFTY)	GROUND SCATTERING CROSS-SECTION/SQ.M.
SLOBE(IRAD)	AVERAGE SIDELOBE GAIN OF RADAR IRAD
SNR	SIGNAL TO TOTAL NOISE RATIO =PT/(PN+PC+POWERJ)
SOJT(ISOJ)	SWITCH ON TIME FOR STAND-OFF JAMMERS
SOJX(ISOJ)	X COORD OF STAND-OFF JAMMER
SOJY(ISOJ)	Y COORD OF STAND-OFF JAMMER
SPLAT(IR)	LATERAL AIRCRAFT SPACING IN RAID IR
SPOTBW(IJTYPE)	MINIMUM BANDWIDTH JAMMER IS CAPABLE OF JAMMING(MHZ)
SQTREP	SQUARE ROOT OF NO. OF REPLICATIONS (SUBROUTINE OUTPUT)
STRAIL(IR)	INTER-FLIGHT SPACING IN RAID IR
T1(IBASE)	TIME TO CAP OR TO INTERCEPTION, EXCL. TAKE-OFF DELAY (SUBROUTINE FCHOOSE)
TBGO(IBASE)	TIME WHEN BASE IBASE IS DUE TO BE FREE OF FIGHTERS QUEUEING FOR TAKE-OFF
TBLAUNCH(IBASE)	TIME BEFORE LAUNCH OF A FIGHTER FROM A BASE
TDELAY	DELAY BETWEEN CALCULATION AND ADOPTION OF AN INTERCEPTION COURSE
TEND	TIME AT WHICH EACH REPLICATION IS TERMINATED
TEV3	TIME AT WHICH THE FIRST RAID IS DETECTED
TF(ICAP,IBASE,IFTY)	TIME FIGHTER TYPE IFTY TAKES TO REACH CAP ICAP FROM BASE IBASE
TFRAME(IFTY)	RADAR FRAME TIME
THILP(IF)	SUBROUTINE LEADPURSUIT: CURRENT ANGLE BETWEEN A RAID'S POSITION VECTOR RELATIVE TO FIGHTER IF & ITS VELOCITY VECTOR
TIME	CURRENT MODEL TIME(MEASURED IN FLOATING-POINT SECONDS)
TIMEM	CURRENT MODEL TIME IN MINUTES
TINT	TIME TO INTERCEPTION,CALCULATED IN SUBROUTINE COLLISION
TINT1(I)	ARRAY OF INTERCEPTION TIMES (SUBROUTINE FFIND)
TMINO(I)	ARRAY OF INTERCEPTION TIMES (SUBROUTINE REATTACK)
TOF(IANG,MTYPE)	TIME OF FLIGHT FOR MISSILE TYPE MTYPE
TONBASE(IBASE)	DELAY BEFORE TAKE-OFF (SUBROUTINE FCHOOSE)
TOTB(IR)	TOTAL NUMBER OF BOMBERS OF RAID IR
TOTJ(IR)	TOTAL NUMBER OF JAMMERS OF RAID IR
TPPT(IPROF,IFTY,IPPT)	TIME OF PROFILE POINT IPPT ON PROFILE IPROF OF FIGHTER TYPE IFTY
TPRINT	TIME AT WHICH EVENT LISTING STARTS
TR(IT,IR)	TIME AT WHICH TRACK IT OF RAID IR STARTS
TRACESTART	TIME AT WHICH FULL TRACE STARTS
TRACEEND	TIME AT WHICH FULL TRACE ENDS
TRESPONSEF(IFTY)	DELAY BEFORE TARGET DATA FROM FIGHTER IS MADE AVAILABLE TO GROUND CONTROL
TRNRATE(IFTY)	FIGHTER TURN-RATE WHEN MAKING A REATTACK
TRESPONSER(IR)	TOTAL REACTION DELAY TO TRACK-CHANGE OF RAID IR; SEE SUBROUTINE RESPACC
TSEC(T)	STATEMENT FUNCTION FOR CONVERTING MIN.SEC TO SEC.
TSTART	TIME AT WHICH EACH REPLICATION STARTS
TTOT(IBASE)	TIME TO CAP OR TO INTERCEPTION, INCL. TAKE-OFF DELAY (SUBROUTINE FCHOOSE)
TTURN	TIME TAKEN FOR FIGHTER TO TURN ONTO AN ATTACK HEADING
TTURNO(I)	ARRAY OF FIGHTER TURN TIMES (SUBROUTINE REATTACK)
TWSSNR(IFTY)	SIGNAL TO NOISE RATIO FOR TRACK WHILE SCAN INITIATION
TXSEC(IANG,1)	RADAR CROSS-SECTION FOR BOMBERS/SELF-SCREENING JAMMERS
TXSEC(IANG,2)	RADAR CROSS-SECTION FOR SPECIALIST ESCORT JAMMERS
TXSECTION	TOTAL RADAR CROSS-SECTION OF RAID AS SEEN BY FIGHTER
TX	X DIRECTION COSINE OF TARGET FROM RADAR
TY	Y DIRECTION COSINE OF TARGET FROM RADAR
TZ	Z DIRECTION COSINE OF TARGET FROM RADAR

VO	INITIAL FIGHTER SPEED IN SUBROUTINES INTCALCONTROL AND COLLISION
VC	CLOSING VELOCITY OF TARGET WITH FIGHTER
VISIDENT(IFTY)	DISTANCE WITHIN WHICH VISUAL IDENTIFICATION OCCURS
VPPT(IPROF,IFTYPE,IPPT)	VELOCITY OF A FIGHTER TYPE AT A PROFILE PT
VR(IT,IR)	VELOCITY OF TRACK IT OF RAID IR
VX	HORIZ. COMPONENT OF FIGHTER SPEED (CLUTTER CALCS.)
VZ	VERTICAL COMPONENT OF FIGHTER SPEED (CLUTTER CALCS.)
VXR(IT,IR)	X-COMP OF VEL OF TRACK IT OF RAID IR
VYR(IT,IR)	Y-COMP OF VEL OF TRACK IT OF RAID IR
WLENGTH	RADAR WAVE LENGTH
WLBYLPI	RADAR WAVELENGTH DIVIDED BY 4 PI.
X	FIGHTER'S INITIAL X-COORD. AT START OF INTERCEPTION COURSE CALCULATION
XBLOSS	NUMBER OF BOMBERS LOST FROM A FLIGHT
XCOMP(IBASE,ICAP)	X DIFFERENCE IN DISTANCE BETWEEN CAP AND BASE
XJLOSS	NUMBER OF JAMMERS LOST FROM A FLIGHT
XPK	EFFECTIVE MISSILE KILL PROBALITY (SUBROUTINE RAIDKILL)
XPT(ICAP,IPT)	X-COORDS OF THE POINTS DEFINING EACH CAP
XR(IT,IR)	X-COORD OF START OF TRACK IT OF RAID IR
XRELMIN	X-COORD. OF INTERCEPTION POINT CORR. TO THE FIGHTER WHICH CAN INTERCEPT MOST QUICKLY (SUBROUTINE FCHOOSE)
Y	FIGHTER'S INITIAL Y-COORD. AT START OF INTERCEPTION COURSE CALCULATION
YCOMP(IBASE,ICAP)	Y DIFFERENCE IN DISTANCE BETWEEN CAP AND BASE
YPT(ICAP,IPT)	Y-COORDS OF THE POINTS DEFINING EACH CAP
YR(IT,IR)	Y-COORD OF START OF TRACK IT OF RAID IR
YRELMIN	Y-COORD. OF INTERCEPTION POINT CORR. TO THE FIGHTER WHICH CAN INTERCEPT MOST QUICKLY (SUBROUTINE FCHOOSE)

ILLUSTRATIVE DATA FILE

FIGURE B1 ILLUSTRATES THE SIMPLE TEST SCENARIO. TWO RAIDS ARE CONSIDERED, WITH PRE-DETERMINED TRACKS AS SHOWN. THE POSITIONS OF THE THREE WARNING LINES, THE TWO CAPS AND THE TWO FIGHTER BASES IN THE SCENARIO ARE ALSO SHOWN. THE MEANING OF MOST OF THE ITEMS IN THE DATA FILE, CONSIDERED IN CONJUNCTION WITH FIGURE B1, SHOULD BE SELF-EVIDENT.

TEST DATA FILE FOR FIGHTER MODEL

```
CONTAINS 2 RAIDS,2 BASES

SECTION 1: HOUSEKEEPING DATA:

0.0           =TIME AT WHICH EVENT LISTING STARTS(SEC)

0             =1/0 IF RAID POSITION DATA IS/IS NOT TO BE OUTPUT

0             =1/0 IF FIGHTER POSITION DATA IS/IS NOT TO BE OUTPUT

0             THIS IS A LIST OF EVENT TYPES NOT TO BE LISTED
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
SECTION 2: RUN DATA

0.0           =START TIME(MIN.SEC)

25.0          =END   TIME(MIN.SEC)

1             =MODE OF RUN (=0/1 IF DETERMINISTIC/STOCHASTIC)

10            =NO. OF REPLICATIONS TO BE CARRIED OUT

987654        =INITIAL SEED VALUE FOR RANDOM NUMBER STREAM

1             =1 IF EVENT TYPE 12 IS THE MEASURE OF FIGHTER EFFECTIVENESS
              =0 IF EVENT TYPE 16 IS THE MEASURE OF FIGHTER EFFECTIVENESS

10.0          =TIME INTERVAL(SEC) BETWEEN RADAR ROUTINE CALLS

5.0           TIME BETWEEN RANGE CHECKS FOR MISSILE LAUNCH RANGE AFTER DETECTION

0.0           TIME AT WHICH DETAILED TRACE STARTS,IF REQUIRED (MINS)

0.0           TIME AT WHICH DETAILED TRACE ENDS,IF REQUIRED (MINS)

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
SECTION 3:RAID DATA

2     RAIDS
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

RAID  1:

1             SERIAL OF CAP MOUNTED IN INITIAL RESPONSE TO THIS RAID

0.0           TIME AT WHICH RAID TRACK STARTS(MIN.SEC)
```

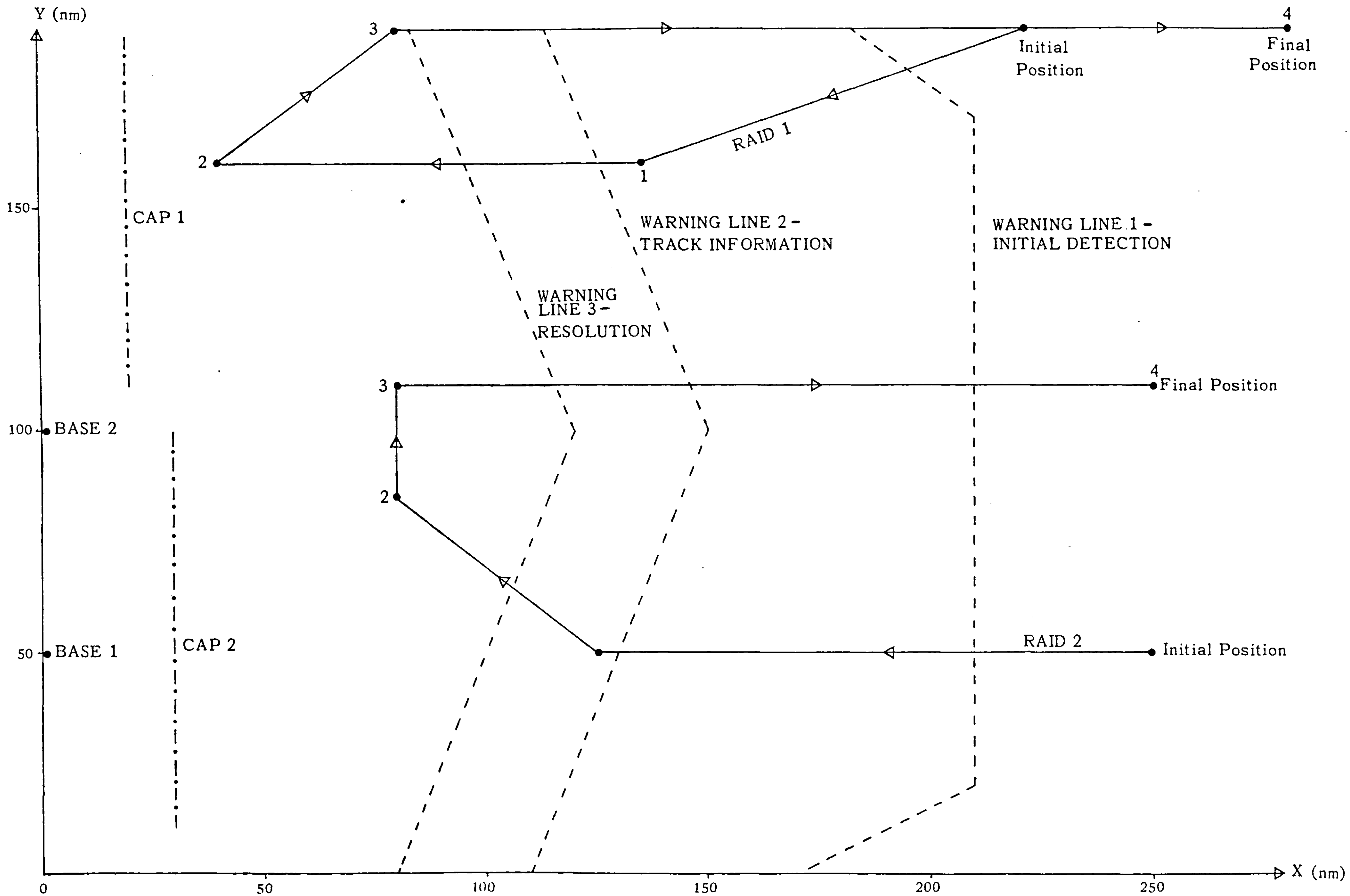


FIGURE B.1: TEST SCENARIO FOR FIGHTER MODEL

```
5      'FLIGHTS'

 500   M INTER-AIRCRAFT SPACING

2000   M INTER-'FLIGHT' SPACING

FORMATION:    -'FLIGHT'-
         1    2    3    4    5

         5    6    5    6    5                BOMBERS/SSJ
         1    0    1    0    1                SPECIALIST JAMMERS

4      TRACK-LEGS

COURSE:       -TRACK-CHANGE-POINT
          0       1    2    3    4

         220     135   40   80  250    X)   COORDS OF
         190     160  160  190  190    Y)   TRACK-CHANGE-PTS(NM)
           9       9    9    9   15         RAID SPEED AT TRACK-CHANGE-PT(NM/MIN)
        2000    2000 2000 2000 2000         ALTITUDE(M)

2      TRACK-CHANGE POINT OF WEAPON-RELEASE(-1 FOR SPOOF RAID)
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

RAID  2:

2             SERIAL OF CAP MOUNTED IN INITIAL RESPONSE TO THIS RAID

10.0          TIME AT WHICH RAID TRACK STARTS(MIN.SEC)

2      'FLIGHTS'

 500   M INTER-AIRCRAFT SPACING

2000   M INTER-'FLIGHT'   SPACING

FORMATION:    -'FLIGHT'-
         1    2

        10   10                               BOMBERS/SSJ
         5    5                               SPECIALIST JAMMERS

4      TRACK-LEGS

COURSE:       -TRACK-CHANGE-POINT
          0       1    2    3    4

         250     125   80   80  250    X)   COORDS OF
          50      50   85  110  110    Y)   TRACK-CHANGE-PTS(NM)
           9       9    9    9   15         RAID SPEED AT TRACK-CHANGE-PT(NM/MIN)
        9000    9000 9000 9000 9000         ALTITUDE(M)

3      TRACK-CHANGE POINT OF WEAPON-RELEASE(-1 FOR SPOOF RAID)
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
SECTION 4: EARLY WARNING DATA
--------------------------------------------------------------------------------

EW LINE 1:INITIAL DETECTION:

4 DATA POINTS

  170  210  210  170      X) COORDINATES(NM)
    0   20  170  200      Y)
--------------------------------------------------------------------------------
```

```
EW LINE 2:TRACK DATA AVAILABLE:

3 DATA POINTS

  110  150  110           X) COORDINATES(NM)
  200  100    0           Y)
--------------------------------------------------------------------------------

EW LINE 3:RAID RESOLVED:

3 DATA POINTS

   80  120   80           X) COORDINATES(NM)
  200  100    0           Y)
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
SECTION 5: CAP AND FIGHTER AIR-BASE DATA

2    =NO.OF CAPS CONSIDERED

     NOTE THAT THE ORDER IN WHICH THE CAP POINTS ARE INPUT DEFINES
     THE ORDER IN WHICH THE PATROLLING FIGHTERS PASS THROUGH THE POINTS
--------------------------------------------------------------------------------

CAP   1:

2    =NO.OF POINTS DESCRIBING THE CAP

 POINT NUMBER
  1       2

 20       20       X)COORDS OF POINTS DESCRIBING THE CAP
110      190       Y)
--------------------------------------------------------------------------------

CAP   2:

3    =NO.OF POINTS DESCRIBING THE CAP

 POINT NUMBER
  1       2       3

 30       30       30       X)COORDS OF POINTS DESCRIBING THE CAP
 10       50      100       Y)
--------------------------------------------------------------------------------

2    =NO.OF FIGHTER BASES CONSIDERED

2    =NO.OF FIGHTER TYPES CONSIDERED

3    =NO.OF READINESS LEVELS CONSIDERED

LEVEL     TIME BEFORE FIGHTERS AVAILABLE FOR TAKE-OFF

1           2  MINS.
2           5  MINS.
3          15  MINS.

--------------------------------------------------------------------------------
```

```
BASE  1:

0    X)COORDS OF BASE(NM)
50   Y)

 1      2                 FIGHTER TYPE

15      0                 NO.INITIALLY ON BASE

 5      0                 NO.AT READINESS LEVEL 1
10      0                 NO.AT READINESS LEVEL 2
 0      0                 NO.AT READINESS LEVEL 3

CAP   POINT OF ACCESS

1     1                   )A CAP CAN BE JOINED ONLY AT ONE OF THE POINTS
2     2                   )WHICH DETERMINE THE CAP

0.30    =CONSTANT TIME BETWEEN FIGHTER TAKE-OFFS(MINS.SECS)
---------------------------------------------------------------------------

BASE  2:

0    X)COORDS OF BASE(NM)
100  Y)

 1      2                 FIGHTER TYPE

 0      30                NO.INITIALLY ON BASE

 0       0                NO.AT READINESS LEVEL 1
 0      15                NO.AT READINESS LEVEL 2
 0      15                NO.AT READINESS LEVEL 3

CAP   POINT OF ACCESS

1     1                   )A CAP CAN BE JOINED ONLY AT ONE OF THE POINTS
2     3                   )WHICH DETERMINE THE CAP

0.30    =CONSTANT TIME BETWEEN FIGHTER TAKE-OFFS(MINS.SECS)
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
SECTION 6: FIGHTER DATA

2    =NO.OF PROFILES CONSIDERED

---------------------------------------------------------------------------

AIRCRAFT TYPE 1

1          =FIGHTER TURN RATE,R (=3*R DEGREES/SEC)

4000       =FUEL RESERVE(LBS) NEEDED TO ATTEMPT AN INTERCEPTION

3500       =CAP HEIGHT(METRES)

PROFILE 1

22000      =INITIAL FUEL CAPACITY ON PROFILE 1(LBS)

  800      =FUEL CONSUMED ON START,TAXI,RUN-UP & ACCELERATE TO CLIMBSPEED
```

6 =NO. OF POINTS AT WHICH THIS PROFILE IS SPECIFIED

PROFILE POINT	TIME(MIN.SEC)	VELOCITY(NM/MIN)	FUEL CONSUMPTION RATE(LBS/MIN)
1	0	0	350
2	2	3	250
3	4	5	150
4	5	6	150
5	6	6.5	125
6	80	6.5	125

PROFILE 2

22000 =INITIAL FUEL CAPACITY ON PROFILE 2(LBS)

1400 =FUEL CONSUMED ON START,TAXI,ETC

5 =NO. OF POINTS AT WHICH THIS PROFILE IS SPECIFIED

PROFILE POINT	TIME(MIN.SEC)	VELOCITY(NM/MIN)	FUEL CONSUMPTION RATE(LBS/MIN)
1	0	0	450
2	2	4	300
3	4	7	250
4	6	10	200
5	80	10	200

AIRCRAFT TYPE 2

1.5 =FIGHTER TURN RATE,R (=3*R DEGREES/SEC)

4000 =FUEL RESERVE (LBS) NEEDED TO ATTEMPT AN INTERCEPTION

5000 =CAP HEIGHT(METRES)

PROFILE 1

21000 =INITIAL FUEL CAPACITY ON PROFILE 1(LBS)

900 =FUEL CONSUMED ON START,TAXI,ETC.

6 =NO. OF POINTS AT WHICH THIS PROFILE IS SPECIFIED

PROFILE POINT	TIME(MINS)	VELOCITY(NM/MIN)	FUEL CONSUMPTION RATE(LBS/MIN)
1	0	0	375
2	2	2.5	200
3	4	4.5	150
4	5	6	150
5	6	7	125
6	80	7	125

PROFILE 2

21000 =INITIAL FUEL CAPACITY ON PROFILE 2(LBS)

1400 =FUEL CONSUMED ON START,TAXI,ETC

5 =NO. OF POINTS AT WHICH THIS PROFILE IS SPECIFIED

PROFILE POINT	TIME(MIN.SEC)	VELOCITY(NM/MIN)	FUEL CONSUMPTION RATE(LBS/MIN)
1	0	0	450
2	2	4	300
3	4	7	250
4	6	8.5	200
5	80	8.5	200

>>

SECTION 7:COMMAND AND CONTROL DATA

2 =INITIAL NO.OF FIGHTERS COMMITTED TO A HOLDING CAP

2 =INITIAL NO.OF FIGHTERS COMMITTED TO A RAID OF UNKNOWN STRENGTH

0.1 THIS IS THE RATIO FIGHTERS:ENEMY AIRCRAFT IN A RAID WHICH THE ALLOCATION PROCESS WILL ATTEMPT TO ACHIEVE WHEN THE STRENGTH OF THE RAID IS KNOWN.

1 CLOSE CONTROL POSSIBLE?(1/0=YES/NO)

1 1 OR 0 RESPECTIVELY IF A FIGHTER UNDER CC CONTINUES ON THIS COURSE OR ADOPTS A L.P. COURSE IF IT ACHIEVES DETECTION BUT NOT BURNTHROUGH

20 INITIAL NO. OF INTERCEPT CONTROLS AVAILABLE(=2*NO. OF INTERCEPT CONTROLLERS APPROXIMATELY)

FIGHTER TARGET REPORTING PARAMETERS:
(I.E. AIR-TO-GROUND COMMUNICATIONS; DEPENDENT ON THE FIGHTER TYPE)

FIGHTER TYPE:

1	2	
10.0	20.0	DELAY IN SEC BEFORE TARGET DATA FROM FIGHTER MADE AVAILABLE
5.0	3.0	S.D. OF ERROR IN REPORTING TARGET POSITION (NM)
0.3	0.5	S.D. OF ERROR IN REPORTING TARGET SPEED(NM/MIN)
3.0	5.0	S.D. OF ERROR IN REPORTING TARGET HEADING (DEG)
5.0	5.0	DISTANCE WITHIN WHICH VISUAL IDENTIFICATION OCCURS(NM)

GROUND TARGET REPORTING PARAMETERS:

5.0	DELAY IN SECS BEFORE TARGET DATA FROM GROUND MADE AVAILABLE
1.0	S.D. OF ERROR IN REPORTING TARGET POSITION (NM)
0.1	S.D. OF ERROR IN REPORTING TARGET SPEED(NM/MIN)
2.0	S.D. OF ERROR IN REPORTING TARGET HEADING (DEG)

TIME DELAYS & ACCURACIES OF INTERCEPTION COURSES FOR VARIOUS CASES:
(I.E. GROUND-AIR COMMUNICATIONS; DEPENDENT ON FIGHTER CONTROL MODE)

CASE	DELAY (SEC)	HEADING ERROR (S.D. DEG.)	DESCRIPTION OF EACH CASE
1	2.0	1.0	INTERCEPTION USING DL+ONBOARD COMPUTER
2	15.0	5.0	BROADCAST CONTROLLED INTERCEPTION
3	1.0	1.0	NORMAL CLOSE CONTROL FROM GROUND
4	2.0	2.0	INTERCEPTION USING AI RADAR

OUT OF BOUNDS LINES LIMITING FIGHTER FLIGHT (MEZ'S FOR EXAMPLE):

1 NO. OF LINES (EACH LIMITED BY 2 POINTS)

LINE	X1	Y1	X2	Y2	
1	200	0	200	200	NM

>>

SECTION 8:RADAR DATA

GENERAL:

250 NM RANGE ABOVE WHICH RADAR/JAMMER INTERACTIONS IGNORED

RADAR PERFORMANCE DATA:

RADAR TYPE(1 TYPE PER FIGHTER TYPE):

1	2	UNIT	
100	10	W	MEAN POWER
1000	2000	W	PEAK POWER
100000	5000	HZ	PULSE REPETITION FREQUENCY
9500	9500	MHZ	MIN CARRIER FREQUENCY
10000	10000	MHZ	NOMINAL CARRIER FREQUENCY
10500	10500	MHZ	MAX CARRIER FREQUENCY (EACH FIGHTER IS ALLOCATED A DIFFERENT CARRIER FREQUENCY WITHIN THE RANGE SPECIFIED FOR ITS TYPE)
0	0	HZ/S	SLOPE OF LINEAR RANGING MODULATION
5.0	5.0	-DB	1-WAY TRANSMISSION LOSS(AFFECTS ALL TRANSMISSIONS)
4.0	4.0	-DB	1-WAY RECEPTION LOSS(AFFECTS ALL RECEIVED SIGNALS)
0	0	-DB	PD ECLIPSING LOSS(AFFECTS TARGET & CLUTTER ONLY)
60	60	DB	THRESHOLD FOR DETECTION OF JAMMING
3.0	3.0	S	FRAME TIME
80	80	DEG/S	SCANNING RATE
30	30	+DB	MAX GAIN(1-WAY)
4	4	DEG	3-DB BEAMWIDTH
30	30	-DB	NOMINAL SIDELOBE GAIN(DOWN ON MAIN BEAM)
60	60	DEG	AZIMUTH SCAN HALF-WIDTH
20	20	DEG	INITIAL LEAD PURSUIT COURSE LEAD ANGLE
1	0		TWS?(1/0=YES/NO)
1	0		DL ?(1/0=YES/NO)
1	1		GROUND SCATTERING CROSS-SECTION/SQ.M. AT NORMAL INCIDENCE(USED IN CLUTTER CALCULATIONS)
1	1		EXPONENT OF SIN(GRAZING ANGLE) IN GROUND CROSS-SECTION FORMULA (USED IN CLUTTER CALCULATIONS)
1	1		SWERLING G-FACTOR
-180	-180	DB	RECEIVER NOISE POWER
1000	100	HZ	RECEIVER BANDWIDTH
10	10	DB	J:N RATIO FOR HOJ INITIATION
5	5	DB	J:N RATIO FOR TERMINATION OF HOJ
2	2	DB	S:J RATIO FOR HOJ INITIATION
2	2	DB	S:J RATIO FOR TERMINATION OF HOJ & AOJ
20	20	DB	SNR FOR TWS INITIATION
10	10	DB	SNR FOR SATISFACTORY RANGE DETERMINATION
-1	-1	DB	SNR FOR INITIAL DETECTION OF TARGET

RADAR ANTENNA GAIN PATTERNS:(1-WAY GAIN IN DB IS GIVEN AS A FUNCTION OF ANGLE OFF-AXIS IN DEGREES FROM 0-180:ANGLES NEED NOT BE UNIFORMLY SPACED AS PROGRAM LINEARLY INTERPOLATES)

RADAR 1:

0.0	2.5	45.0	90.0	135.0	180.0	ANGLE
30.	27.	15.0	0.0	0.0	0.0	GAIN DB

RADAR 2:

0.0	2.5	45.0	90.0	135.0	180.0	ANGLE
30.	27.	15.0	0.0	0.0	0.0	GAIN DB

ENEMY RADAR CROSS-SECTIONS:(FORMAT AS FOR GAIN PATTERNS)

0 =0/1 IF DATA IN SQ M/DB REFERRED TO 1 SQ M

ENEMY AIRCRAFT TYPE 1:BOMBERS/SSJ:

0.0	45.0	90.0	135.0	180.0	ANGLE
15.0	15.0	15.0	15.0	15.0	XSECTION

ENEMY AIRCRAFT TYPE 2:SPECIALIST JAMMERS:

0.0	45.0	90.0	135.0	180.0	ANGLE
10.0	10.0	10.0	10.0	10.0	XSECTION

>>

SECTION 9:JAMMER DATA

GENERAL:

9000 MHZ LOWEST RADAR FREQUENCY CONSIDERED IN THE MODEL
10500 MHZ HIGHEST RADAR FREQUENCY CONSIDERED IN THE MODEL

2 NO. OF (EQUAL)SUBBANDS INTO WHICH ABOVE FREQUENCY RANGE DIVIDED FOR JAMMING PURPOSES

2 NO. OF TYPES OF ENEMY JAMMER(EXCLUDING SOJ'S:CURRENTLY COMPRISES SSJ'S AND SPECIALIST ESCORT JAMMERS)

3.0 -DB POLARISATION LOSS(AFFECTS RECEPTION & TRANSMISSION BY ALL JAMMERS EQUALLY)

SPECIFIC JAMMER CHARACTERISTICS:

JAMMER TYPE 1:NON-SPECIALIST(EG BOMBER WITH ECM POD):
- - - - - - - -

50 MHZ:BANDWIDTH FOR SPOT-JAMMING

0 =1/0 IF JAMMER ABLE/UNABLE TO DETECT & RESPOND TO SCANNING RADAR DURING PAINT ONLY

BAND	POWER ALLOCATED TO BAND(W):
1	1000
2	1000

-100 DB:THRESHOLD POWER LEVEL FOR JAMMER TO DETECT RADARS

JAMMER RECEIVER POLAR DIAGRAM:ABSOLUTE:(FORMAT AS FOR RADAR GAINS ABOVE):

0.0	45.0	90.0	135.0	180.0	ANGLE
3.0	0.0	-3.0	0.0	3.0	GAIN DB

JAMMER TRANSMITTER POLAR DIAGRAM

0.0	45.0	90.0	135.0	180.0	ANGLE
3.0	0.0	-3.0	0.0	3.0	GAIN DB

```
JAMMER TYPE 2:SPECIALIST
- - - - - - - -
50     MHZ:BANDWIDTH FOR SPOT-JAMMING

1      =1/0 IF JAMMER ABLE/UNABLE TO DETECT & RESPOND TO SCANNING RADAR
       DURING PAINT ONLY

BAND   POWER ALLOCATED TO BAND(W):
1      2000
2      2000

-100    DB:THRESHOLD POWER LEVEL FOR JAMMER TO DETECT RADARS

JAMMER RECEIVER POLAR DIAGRAM:ABSOLUTE:(FORMAT AS FOR RADAR GAINS ABOVE):

0.0    45.0  90.0  135.0 180.0                                    ANGLE
3.0     0.0  -3.0    0.0   3.0                                    GAIN DB

JAMMER TRANSMITTER POLAR DIAGRAM

0.0    45.0  90.0  135.0 180.0                                    ANGLE
3.0     0.0  -3.0    0.0   3.0                                    GAIN DB

STAND-OFF JAMMERS:
- - - - - - - - - -

2      SOJ'S

LOCATIONS(ASSUMED STATIONARY):

SOJ1  SOJ2
 200.  100.     X NM
 200.    0.     Y NM
 0.0   0.0      SWITCH-ON TIME(MIN.SEC)

TRANSMITTER POWERS(ASSUMED ISOTROPIC FOR THE TIME BEING):

SOJ1  SOJ2
 5000  5000     WATTS  BAND 1 (9000-9750 MHZ)
10000 10000     WATTS  BAND 2 (9750-10500 MHZ)

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
SECTION 10:MISSILE DATA:

2      AIR-TO-AIR MISSILE TYPES

MISSILE MODES OF OPERATION(1/2=RADAR/INFRA-RED):

MISSILE1 MISSILE2
    1       2

FIGHTER MISSILE PAYLOADS:

FIGHTER1 FIGHTER2
   4        1              MISSILE1
   4        5              MISSILE2

FIGHTER 1ST PRIORITY MISSILE TYPE IN DIFFERENT SITUATIONS:

FIGHTER1 FIGHTER2
   1        1            FORWARD HEMISPHERE ATTACK
   2        1            REAR    HEMISPHERE ATTACK
```

MISSILE LETHALITIES(%):

MISSILE1	MISSILE2		
80	10	CLEAR	)
40	10	HOJ	)FORWARD HEMISPHERE ATTACK
20	10	AOJ	)
60	50	CLEAR	)
30	50	HOJ	)REAR HEMISPHERE ATTACK
15	50	AOJ	)

MISSILE LAUNCH SUCCESS ZONES(FORMAT SAME AS FOR AERIAL GAINS & OTHER POLAR DIAGRAMS:UNITS KM FOR RANGE & SEC FOR TIME)

MISSILE1:LAUNCH SUCCESS RANGE (MAX):

0.0	180.0	ANGLE DEG
10.0	10.0	RANGE KM

MISSILE1:TIME OF FLIGHT(FROM MAX RANGE):

0.0	180.0	ANGLE DEG
10.0	10.0	TOF SEC

MISSILE1:LAUNCH SUCCESS RANGE (MIN):

0.0	180.0	ANGLE DEG
2.0	2.0	RANGE KM

MISSILE2:LAUNCH SUCCESS RANGE (MAX):

0.0	180.0	ANGLE DEG
5.0	5.0	RANGE KM

MISSILE2:TIME OF FLIGHT(FROM MAX RANGE):

0.0	180.0	ANGLE DEG
5.0	5.0	TOF SEC

MISSILE2:LAUNCH SUCCESS RANGE (MIN):

0.0	180.0	ANGLE DEG
1.0	1.0	RANGE KM

0.9 FACTOR BY WHICH ACTUAL LAUNCH RANGE REDUCED FROM ABOVE MAXIMA

70.0 DEG:ENEMY ASPECT ANGLE AT COLLISION ABOVE WHICH AN ATTACK IS DEFINED TO BE REAR HEMISPHERE

PART 3

A MATHEMATICAL ANALYSIS

OF

RAID INDIRECT ROUTING

CONTENTS OF PART 3 - A MATHEMATICAL ANALYSIS OF RAID INDIRECT ROUTING

1. INTRODUCTION

Overview

The work described here developed from DOAE Study 250 (DOAE Report R7707), which investigated the influence of sensors on fighter intercept capability as one aspect of air defence. In order to analyse the effect of the major sensor parameters certain simplifying assumptions were made in Study 250. In particular, it was assumed that:

(i) raids attack their targets along direct tracks

(ii) the information provided by the Ground Control sensors is accurate.

This part of the thesis investigates the problems raised by relaxing assumption (i), i.e. we examine the influence of bomber tactical routing on the intercept capability of fighters scrambled from ground alert. In considering raid indirect routing it will be assumed that the information provided to the fighters by the sensors through Ground Control (GC) is accurate. Two complementary approaches may be adopted. Mathematical analysis may be employed to give a basic understanding of the many elements involved, while a computer simulation - the fighter model - may be used to study the more complex operational aspects.

Scope

This study considers a single fighter base defending a single offset target against attack from a concentrated (point) raid. The analysis is primarily geometric in nature, and as such requires a pre-defined coordinate system. This could be derived, for example, as in Figure 1.1. We take the fighter base providing the fighter defences at the origin of coordinates, while the y-axis represents a 'target axis.' Fighter interceptions are regarded as successful only if they are achieved before the raid penetrates the target axis, with attacks coming from the hemisphere $x > 0$. This introduces a form of 'area defence' and eliminates the need to define an expected target system, or to specify which fighter bases are likely to defend particular targets. With this convention we may take the raid target to be the intersection of the raid track with the target axis. In the numerical examples considered later we take this point to be half-way between the fighter base and its nearest neighbour along the target

Part 3 - p.2.

Last sentence, para. 2, should read:

The raid track-change point is assumed to be pre-planned by the attacker; thus evasive manoeuvre in response to fighter attack is not considered here, but can be investigated using the fighter model (after suitable modification).

axis (see Figure 1.1). The distance perpendicular to the target axis at which detection by GC sensors first occurs is called the warning distance. A number of the values of warning distance considered in the examples presented in Chapter 2 are sufficiently small so as to correspond to low-level attacks, at high subsonic speed. It can be seen from the assumptions described here, namely:

(i) single fighter base;
(ii) point raid;
(iii) 'area defence' of a target axis;
(iv) maximum target offset along this axis;
(v) low-level attacks;

that the scenarios considered in the numerical examples presented in Chapter 2 are in a number of respects pessimistic for the fighter defences. Favourable conclusions concerning the viability of the fighter defences which can be drawn under these circumstances are then likely to hold also under more favourable conditions for fighter operations.

The raid is assumed to fly one feint leg, at the end of which it heads directly for its target. While the analysis is in principle no more complicated by dropping this assumption, in the case of multiple raid feint tracks the complexity of fighter and raid interactions involved is such that the fighter model would be better suited to such a study. Interception is represented by 'collision', the coincidence of the fighter and the raid on their respective tracks, with no allowance made for offsets for forward or rear engagements, or for reattacks. Hence the planned point of intersection of the fighter and raid tracks is coincident with the expected raid position at that time and is not translated ahead of or behind the raid, although this could give the fighter a greater kill probability in a forward or rear hemisphere attack respectively. Further, no attempt is made to assess the likelihood for any given fighter and raid geometrical configuration of the fighter being able to carry out further attacks after its initial attempted interception. The position, speed and course of the raid at first detection, and at its track-change, are assumed correctly estimated by Ground Control. The raid track-change point is assumed to be pre-planned by the attacker; thus evasive manoeuvre in response to fighter attack is not considered here but can be investigated using the fighter model (Part 2 of the thesis).

Following initial detection of the raid we assume a fixed delay for sensor and GC processing and transmission of the information, followed by a reaction delay representing the readiness level of the fighters. It is further assumed that, during these delays, the raid is identified as hostile and approaching (i.e., does not consist of stand-off jammers, patrolling outside defended airspace and emitting noise or deception jamming in support of the attacking raids - see Part 2) and is in sufficient strength to warrant fighter scramble. We also impose a delay between each fighter scrambled for take-off. There is no limitation assumed in the analysis on the number of fighters at the base, although in the numerical examples presented there is taken to be at most twenty.

The details of the calculations are given in Appendices A and B, while Chapter 2 illustrates their application in a set of simple but significant examples.

Analytical Method

The mathematical approach is an extension of that used in DOAE Study 250. The maximum allowable fighter take-off delay for successful interception is used as a measure of effectiveness of the fighter defences and most results are expressed in terms of this parameter. In the numerical examples, we also present results in terms of the number of interceptions achieved before the raid reaches its target. We had originally hoped to obtain a full understanding of these aspects of fighter operations by running the fighter model but it soon became clear that its results could not easily be interpreted. The mathematical analysis presented here was therefore derived partly to explain the fighter model results.

We consider three categories of interception and four scramble policies (i.e. criteria under which fighters are scrambled for take-off); these may then be studied separately for different feint tracks and warning distances.

Categories of Interception

(i) Interception on the feint leg.

(ii) Interception on the second leg of the track by fighters reallocated from expected interceptions on the feint leg.

(The penalty for premature fighter scramble may also be studied in this category. This corresponds to a fighter scrambled against the feint leg but unable to react to the track-change in time to intercept the raid on its second leg, whereas it could have intercepted had it not scrambled until after the track-change).

(iii) Interception on the second leg of the track by fighters scrambled to intercept after detection of the track-change by the warning sensors.

Scramble Policies

I Scramble is continued as long as interception is expected to occur before the target axis.

II Scramble is continued as long as fighters possess a geometrically feasible interception course.

III The feint leg is ignored and fighters fly along the target axis to a patrol position located directly over the target.

IV Scramble is continued as long as fighters can expect to intercept within a specified time (or distance) from the fighter base.

Because of the underlying assumption that the raid penetration is carried out at low-level, which places severe constraints on the upper speed limit of both the enemy raids and the fighters scrambled from Ground Alert, it is not unreasonable to approximate the fighter and raid flight profiles by constant speed. While the complete flight profile of a raid may involve a number of periods of flight at different heights and speeds, we consider here only that period during which it is vulnerable to attack by fighters scrambled from Ground Alert - say within 270 nm. of the target axis (Figure 1.1). We assume that during this final phase the raid flies at its maximum subsonic speed at low-level. (Note that the warning distance achieved is dependent on raid height). Conversely, in general, fighters will have a sufficient distance to travel to their expected interception points such that they too can be assumed to have settled to a steady speed during the interception. (The slight delay between take-off and adoption of the planned interception course and speed can be subsumed into the reaction delay representing fighter readiness level). Finally, because of the assumed accuracy and timeliness of the GC information, we can ignore in this initial analysis the fighters' radar detection performance. Thus, it is assumed that, if an airborne fighter does not detect the raid track-change, it is nevertheless informed by Ground Control, after an appropriate processing delay, of its new interception course against the second track-leg. Further, it is assumed that the reaction delay by fighters which are operating on GC information is equal to the reaction delay to the track-change by fighters which have already detected the raid and are operating autonomously. In a more detailed analysis one could consider different reaction delays to the track-change for fighters operating under close control and fighters operating autonomously. This delay could also depend on the position and velocity of the fighter relative to the raid at the track-change point. We mention here that the fighter model has been used to investigate some of the more complex operational aspects of the problem. These include different capabilities for fighters operating under close control and fighters operating autonomously, and the assessment of fighter intercept effectiveness in the face of raid indirect routing, under both clear and Electronic Counter Measures (ECM) conditions. It can also be used to study realistic fighter velocity and fuel consumption-rate flight profiles, i.e., piecewise-linear graphs of speed, and fuel consumption rate, as a function of time.

Notation

The raid track geometry is illustrated in Figure 1.2. The 'target', i.e. the intersection of the second track-leg with the target axis, is at the point T with coordinates $(0, y_T)$, where we may take

$$y_T \geqslant 0 \quad . \qquad (1.1)$$

Each feint track has an associated feint angle ψ, the angle between the x-axis (i.e. the perpendicular to the target axis) and the feint track, with positive feint angles measured clockwise. We take

$$\frac{-\pi}{2} < \psi < \frac{\pi}{2} \quad . \tag{1.2}$$

Given the feint angle ψ, the feint track is fully defined by specifying its point of intersection F with the target axis. It is convenient to define this point in relation to the raid target T, as shown in Figure 1.2, so tht we specify y_M to be the position relative to T at which the feint line crosses the y-axis. In Chapter 2 we consider families of feint tracks passing through a fixed 'pivot point' at a distance x_T from the target, measured perpendicular to the target axis. From Figure 1.2, the quantities y_M and x_T are related by

$$x_T = y_M \cot \psi,$$

or

$$y_M = x_T \tan \psi. \tag{1.3}$$

The track-change angle θ is the angle between the feint track and the second track-leg, defined as shown in Figure 1.2.

There are six configurations of the raid feint track and direct track-leg to be considered, as shown in Figure 1.3.

Case (1). Raid feints north of the target, $\psi \geqslant 0$.
Case (2). Raid feints north of fighter base and south of target, $\psi \geqslant 0$.
Case (3). Raid feints south of fighter base, $\psi \geqslant 0$.
Case (4). Raid feints north of target, $\psi < 0$.
Case (5). Raid feints north of fighter base and south of target, $\psi < 0$
Case (6). Raid feints south of fighter base, $\psi < 0$.

Case (1) represents the geography underlying the numerical examples of the next Chapter and is shown in more detail in Figure 1.2. The orientations of the feint angle ψ and the track-change angle Θ are defined by this case.

The raid travels at a constant speed U and is first detected at the point S, with coordinates (x_o, y_o), where x_o is the warning distance of the raid. Assuming that the feint is still in force when $x = x_o$, then

$$y_o = y_T + y_M - x_o \tan\psi \quad . \tag{1.4}$$

The raid changes track (instantaneously) at the point C, which is defined either by the track-change angle Θ or the feint time t_s, where t_s is the time spent on the feint track after initial detection. Interceptions are defined to be successful only if they occur before the raid penetrates the target axis, so that we do not need to specify the raid track after it reaches the target.

Following initial detection of the raid there is taken to be a fixed delay D_p for Ground Control processing and transmission of the sensor information, followed by a further reaction delay D_r representing the readiness level of the fighters. There is also a delay D_s between each fighter take-off, after which fighters fly with constant speed V. If for simplicity we suppose that the first fighter scrambled also suffers the delay D_s, then the total delay before take-off, D, for the I-th fighter scrambled is given by

$$D = (D_p + D_r) + I.D_s \quad (I = 1,2,3,.....). \qquad (1.5)$$

As the fighter takes off the raid, having travelled a distance UD, is then at the point R with coordinates (x_1,y_1), where

$$x_1 = x_o - UD \cos\psi ,$$

$$y_1 = y_o + UD \sin\psi . \qquad (1.6)$$

The fighter heading is the angle ϕ made by its track with the positive x-axis, as shown. It is assumed that whether or not a fighter is capable of detecting the raid track-change on its own radar, it adopts if practicable its new interception course against the second track-leg after a delay D_c with respect to the time of the track-change; during this period it continues on its original course.

Note: we will find it convenient in the Appendices to use the parameters p and q, defined by

$$p = (y_M + y_T) \cos\psi = x_o \sin\psi + y_o \cos\psi \qquad (1.7)$$

and

$$q = /y_M \cos\psi/ \quad , \qquad (1.8)$$

where /p/ and q are the shortest distances from the origin and the target respectively to the feint track (see Figure 1.2).

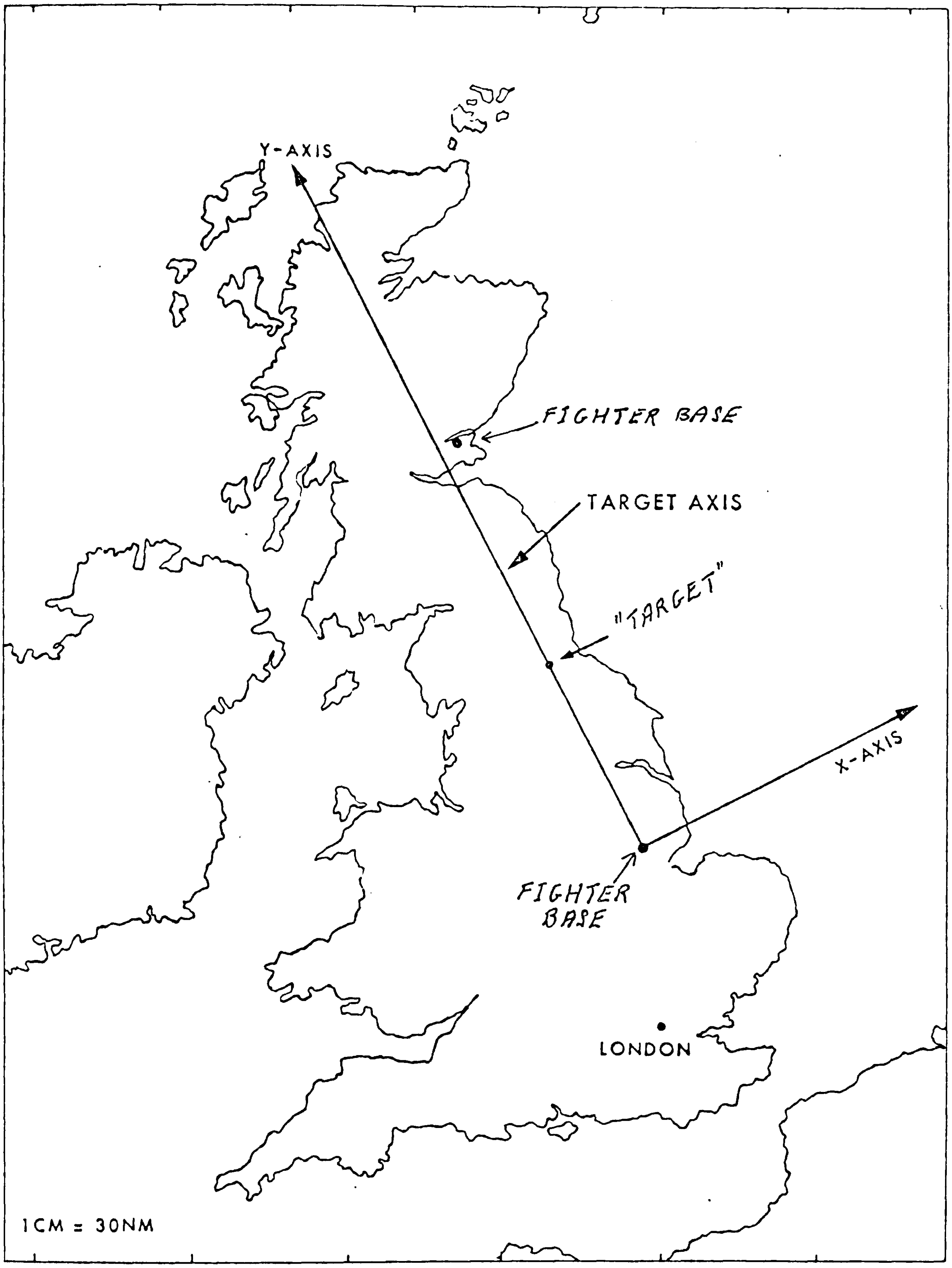


FIGURE 1.1: TARGET AXIS AND CO-ORDINATE SYSTEM

S = raid position when first detected
R = " " " fighter scrambles
C = " track-change point
T = raid 'target'
ψ = feint angle ≥ 0
θ = track-change angle ≥ 0
t_s = feint time
x_0 = warning distance

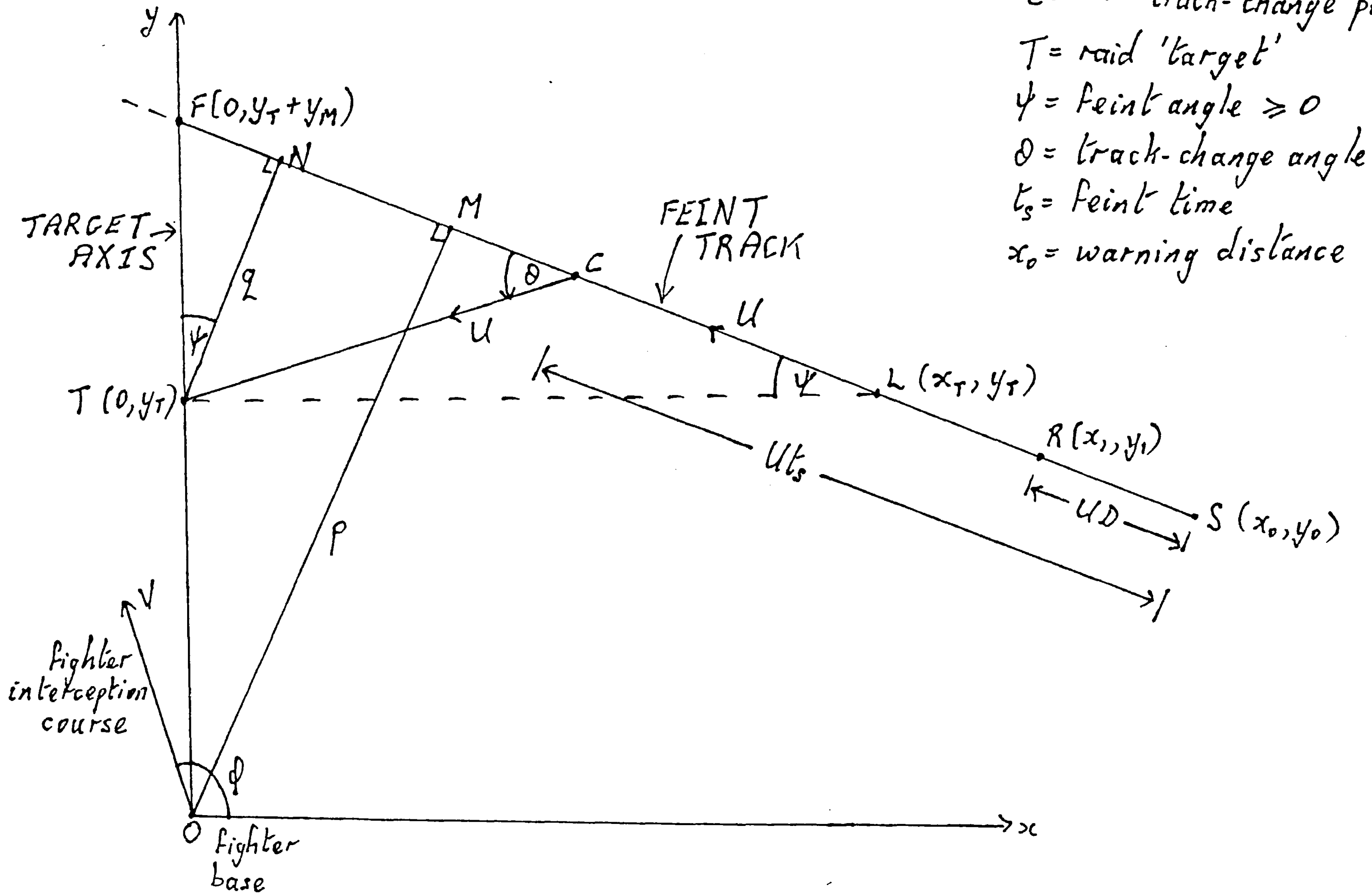


Figure 1.2. Raid and Fighter Geometry (case (1))

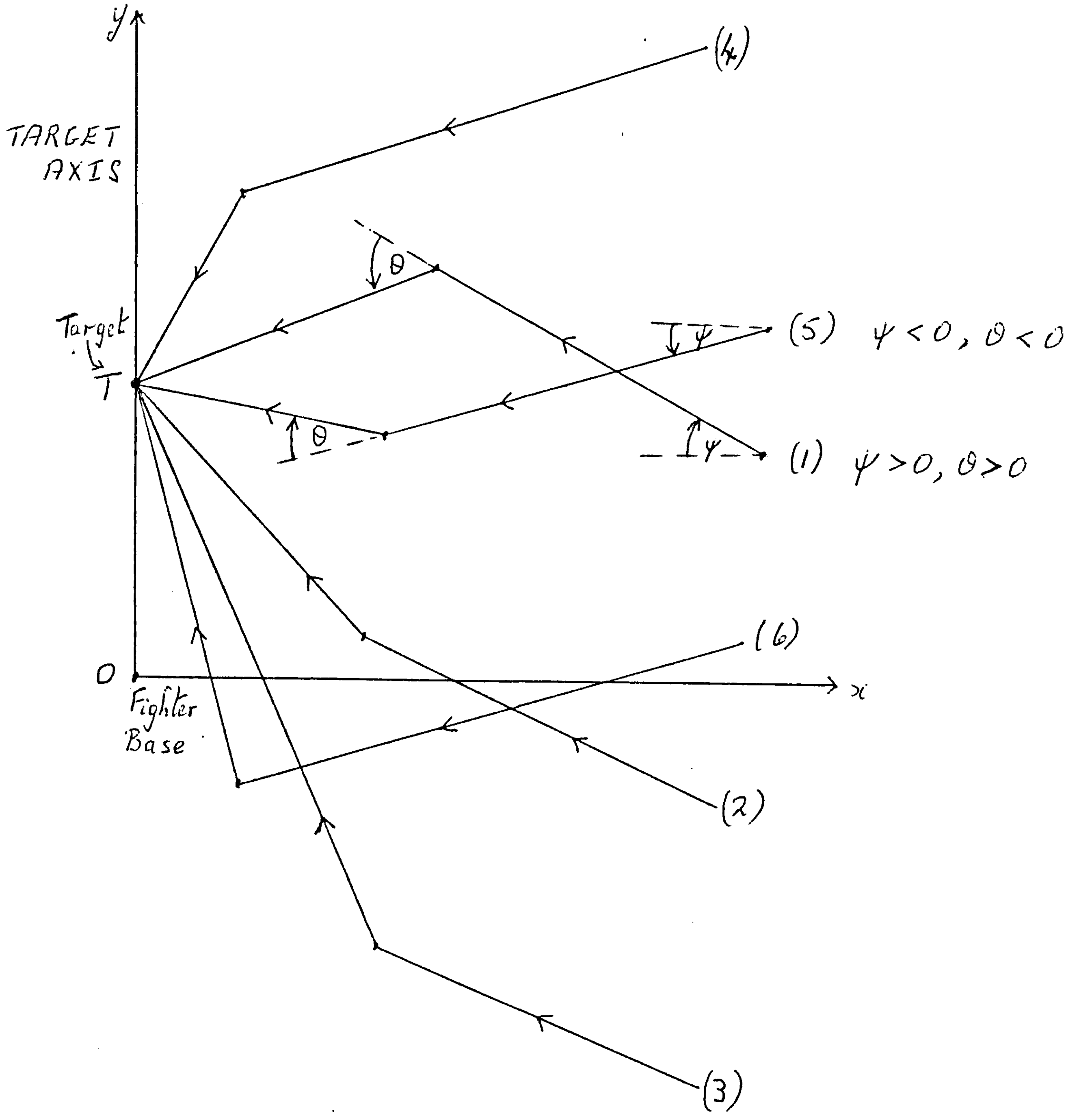


Figure 1.3 Raid Track Geometry - six Cases

2. EXAMPLE

Introduction

To provide a simple application of the analysis of raid indirect routing, we study a special case which, nevertheless, is of considerable interest and practical significance. A scenario is chosen which corresponds geographically to case (1). Furthermore, fighters are assumed to fly at the same speed as the raid and are assumed able to respond immediately to the track-change, which is taken to be at right angles to the feint track (Figure 2.1). It is easily seen from what follows that equation (1.4) relates the maximum allowable take-off delay for possible interception to the corresponding number of interceptions, so that the measure of effectiveness can be taken as the number of interceptions achieved before the raid reaches the target. This is shown as a function of the feint angle, ψ, for a range of realistic warning distances. Each raid track is assumed to pass through a fixed point at a distance $x_T = 270$ nm. from the target, measured perpendicular to the target axis. As mentioned earlier, the fighter model was also used in this study and generally gave very good agreement with the analytical approach; for simplicity the results of the fighter model runs are presented for one particular case only, in Figure 2.15.

Note that a feint away from the fighter base under consideration (at the origin of coordinates) implies a feint towards its nearest neighbour, but a pessimistic case is taken in which no fighters are available from this base to attack the raid. The other simplifying assumptions of this example may be summarised as follows.

(i) Case (1). We consider only feint angles ψ such that

$$0 \leqslant \psi < \frac{\pi}{2} . \tag{2.1}$$

(ii) Track-change angle. For a given feint angle the raid is assumed to abandon its feint when it is closest to the target, i.e.,

$$\Theta = \frac{\pi}{2} . \tag{2.2}$$

(iii) Fighter reaction delay. It is assumed that whether or not a fighter is capable of detecting the raid track-change on its own radar, it is informed of its new intercept course by Ground Control and adjusts its own track immediately, so that

$$D_c = 0 . \tag{2.3}$$

(iv) Equal-speed. We take the fighter speed equal to the raid speed, viz.

$$V = U \ . \tag{2.4}$$

Under these assumptions, the results of the general analysis in Appendices A and B reduce to the following simple form. Only scramble policies I-III are considered; for realistic values of the fighter endurance, policy IV does not restrict the number of possible interceptions in the numerical examples presented.

Scramble Policies (Appendix A)

Here D denotes the maximum acceptable delay between initial GC detection and take-off such that a fighter will scramble under each of the four scramble policies considered in Appendix A.

Policy I (Equation (A12))

$$D = \frac{1}{U} (x_o \sec\psi - (y_T + y_M)) \ . \tag{2.5}$$

Policy II (Equation (A16))

$$D = \frac{1}{U} (x_o \sec\psi - (y_T + y_M) \sin\psi) \ . \tag{2.6}$$

Policy III

If detection takes place on the feint leg, then from (A22)

$$D = \frac{1}{U} (x_o \sec\psi - y_T + y_M (\cos\psi - \sin\psi)). \tag{2.7}$$

Now, the condition that detection occurs on the feint leg is given by

$$x_o > y_M \cos\psi \sin\psi \ . \tag{2.8}$$

If (2.8) does not hold then the raid is first detected by GC after its track-change, in which case, from (A24), D is given by

$$D = \frac{1}{U} (x_o \operatorname{cosec}\psi - y_T) \ . \tag{2.9}$$

Categories of Interception (Appendix B)

Categories (i) and (ii) - Interception by fighters scrambled against the feint leg

Given that a fighter has scrambled onto an accurate interception course against the feint leg, it may then be unable to intercept because of the raid track change. From Figure 1.2, it can be seen that those airborne fighters planning to intercept the raid beyond the track-change point, N, can do so only if they are within a distance /NT/ = $y_M \cos\psi$ of the target when the raid changes track. The maximum take-off delay D for a successful interception is such that scramble occurs when the raid reaches a point at a distance from F equal to

$$y_M \sin\psi + (y_T^2 - y_M^2 \cos^2\psi)^{1/2} \ . \tag{2.10}$$

Hence the required expression for D is

$$D = \frac{1}{U} \left(x_0 \sec\psi - (y_T^2 - y_M^2 \cos^2\psi)^{1/2} - y_M \sin\psi \right). \qquad (2.11)$$

Equation (2.11) holds as long as the scramble limitation under policy II does not come into effect, i.e. as long as

$$y_M < y_T \; . \qquad (2.12)$$

In the Figures presented later, for completeness the graph of equation (2.11) is shown extended beyond its intersection with that of scramble policy II at the point $y_T = y_M$, until it meets the graph of scramble policy III, at the point $y_T = y_M \cos \psi$.

Category (iii) - Interception by fighters scrambled against the second track-leg

All interceptions carried out under scramble policy III may be regarded as in this category, with the maximum allowable take-off delay D given by (2.7) and (2.9). Under the other scramble policies it is assumed that, if the information processing delay D_p has elapsed between first detection of the raid by GC and its track-change, then there is no further such delay imposed on fighters scrambled after the track-change - merely the delay D_r representing their readiness level. In this case, from (B38) the maximum take-off delay D for interception (relative to the time of the track-change) is given by

$$D = \frac{1}{U} (y_M \cos\psi - y_T + U. D_p) \; . \qquad (2.13)$$

The relationship between take-off delay and corresponding number of interceptions is not given by (1.5) but by

$$D = D_r + I. D_s \quad (I = 1, 2, 3, ...). \qquad (2.14)$$

If the raid changes track before the information processing delay D_p is complete (or of course if it is not detected until after the track-change), then the maximum allowable take-off delay is again given by equations (2.7) and (2.9).

Numerical Inputs

The numerical inputs used in the specific examples presented are summarised in Table 2.1. The fighter base is at the point shown in Figure 1.1, with the target (i.e. the point of intersection of the second track-leg with the target axis) halfway between this and the nearest fighter base along the target axis. All feint tracks pass through a fixed point at a distance x_T = 270nm. from the target, measured perpendicular to the target axis. The analysis first assumes that there are no limitations on the number of fighters available at the base; these results are presented in Figures 2.2-2.12. A cut-off is then applied, for a maximum availability of 20 fighters; the corresponding results are presented in Figures 2.13-2.16. (The maximum number of fighters available for scramble from a single fighter base is not a crucial factor in this analysis).

Results - no limitations on the number of fighters available (Figures 2.2-2.12)

Four warning distances are considered, ranging from a minimum of 198nm (at which no interceptions are possible, in these examples, against a direct (perpendicular) raid track) to an assumed maximum value of 360nm. For each warning distance the graphs show the number of interceptions possible as a function of the feint angle, for each of the three scramble policies. Table 2.2 gives a key to the notation of Figures 2.2-2.12. The number of interceptions achieved under policies I and II by fighters scrambled to intercept the feint leg are shown separately from those achieved by fighters scrambled to intercept the second leg. Under policy III fighters are always scrambled to intercept the second track-leg.

Discussion

Under scramble policy I all fighters scrambled to intercept the feint leg achieve successful interceptions. No interceptions on the feint leg are possible with a warning distance of only 198nm(i.e. curve A_I does not exist). If the warning distance is increased to 270nm, 16 interceptions are achieved against a direct attack, perpendicular to the target axis; this falls sharply to zero interceptions at a feint angle of 17^o. Figure 2.5 shows the enormous improvement if the warning distance is increased still further to 360nm.

As the angle of feint increases the time spent by the raid on the second, direct track-leg increases. For feint angles larger than 35^o, interceptions by fighters scrambled after the track-change become possible. As can be seen from Table 2.2, the graph of the number of such interceptions, as a function of the feint angle, is composed of three parts. For raids detected at least 5 minutes before the track-change the processing delay at the track-change is ignored and curve C applies. Otherwise, the situation is identical with that described by scramble policy III, so that curves B_{III} and B'_{III} apply.

From Figure 2.2, no interceptions are possible against a direct track so that a feint can only be positively disadvantageous to the raid. Figures 2.3-2.5 show that under the 'conservative' scramble policy, with a reasonably large warning distance there is a range of feint angles which give considerable advantage to the raid over a direct track (i.e., a feint angle of 0^o). Note that in Figure 2.5 the two curves A_I and C overlap. Thus there is a range of feint angles in which interceptions are possible both by fighters scrambled to intercept the feint leg and by fighters scrambled to intercept the second track-leg.

Figures 2.6-2.9 show the corresponding results for scramble policy II. The curves giving the number of interceptions achieved by fighters scrambled after the track-change are unaltered. The difference between the two policies lies in the number and success of fighters scrambled to intercept the feint leg. The number of interceptions achieved increases rapidly as the feint angle increases, until the scramble limitation is reached. For example,

with 198nm warning distance the number of interceptions increases from zero to nine at a feint angle of 22°, then decreases to zero again at 32°. The complete scramble limitation curves are shown, together with the extension of curve B_{II} to meet B_{III}, the interception limitation under scramble policy III.
These show, where appropriate, the number of fighters scrambled under this policy which fail to intercept, and the number of extra interceptions which could be achieved by a suitable switch to policy III. Again, for larger warning distances there is a range of feint angles in which interceptions are possible both by fighters scrambled to intercept the feint leg and by fighters scrambled after the raid track-change. Finally, Figures 2.10-2.12 give the results achieved under scramble policy III. These are effectively the best possible, corresponding to fighters scrambled to a holding Combat Air Patrol position (CAP), regardless of the direction of the feint track. The CAP is ideally situated, being directly over the target.

Results - maximum availability of 20 fighters (Figures 2.13-2.16)

These results are derived by adding the various components of the fighter intercept capability in Figures 2.2-2.12, and then imposing a cut-off corresponding to a maximum availability of 20 fighters. For each of the four warning distances considered the graphs show the number of interceptions possible, as a function of the feint angle, for each of the three scramble policies, with a limit of 20 fighters available on the base.

The corresponding fighter model results for one of the examples, namely a warning distance of 234 nm, are presented in Figure 2.14 and shown by open circles. As can be seen, the fighter model generally gives excellent agreement with the mathematical analysis. The only anomalous model result, for a feint angle of 30° under scramble policy II, is itself interesting:- it is due to the numerical technique used in the fighter model to calculate the expected interception points being too coarse to generate all the scrambled fighters; this has since been corrected.

Discussion

If the warning distance is only 200nm indirect routing is a positive disadvantage to the raid, since no interceptions are achieved against a direct track (a feint angle of 0°). Under the conservative scramble policy I no fighters are scrambled against the feint leg, while with a feint angle of more than 35° some interceptions are achieved by fighters scrambled after the raid track-change. These rise to a maximum of 13 interceptions with a feint angle of 53°. Under scramble policy II a maximum of 9 interceptions is achieved against the feint leg, at a feint angle of 22°; for larger feint angles the number of fighters scrambled then limits the number of interceptions. Finally, under scramble policy III the number of interceptions rises to 20 as the feint angle increases to 35°, then decreases gradually to zero again.

If the warning distance is increased to 234 nm 8 interceptions are possible against a direct track. Under scramble policy I this number rapidly decreases to zero interceptions on the feint leg as the feint angle rises, while under policy II it increases to 18 interceptions at a feint angle of 22°. At larger feint angles the scramble limitation comes into effect. The maximum number of interceptions achieved by fighters scrambled after the track-change is 19, against a feint angle of 63°. With this warning distance, if Ground Control (GC) adopts scramble policy I there is a wide range of feint angles - between 0° and 45° - which are advantageous to the raid. For angles between 8° and 35°, no interceptions at all are possible. Under policy II a feint is only slightly advantageous to the raid, for a narrow band of feint angles between 33° and 44°. If GC adopts scramble policy III there is a range of feint angles of almost 40° in which all 20 fighters scrambled can intercept. In general, under policies II and III more interceptions are possible if the raid takes an indirect route than if it flies direct to the target. With a warning distance of 270 nm, similar conclusions can be drawn. Under scramble policy I any feint angle between 0° and 57° is advantageous to the raid, while for angles between 17° and 35° no interceptions are achieved. Under scramble policy II a feint is only slightly advantageous to the raid, for a band of feint angles between 35° and 55°, while at angles only slightly outside this band all 20 fighters can intercept. Finally, under scramble policy III, all 20 fighters achieve interceptions against most feint angles.

If the sensors can achieve a warning distance of 360 nm against the raid, only if the fighters adopt scramble policy I is a feint advantageous to the raid. The number of interceptions can be decreased from 20 to a minimum of 12, at a feint angle of 35°. Under scramble policies II and III all 20 fighters intercept, regardless of the feint angle.

Conclusions

Ample warning distance tends to negate any possible advantage to the raid of indirect routing. For a target offset by about 100 nm from the fighter base, and for warning distances between 200 nm and 300nm, the fighter intercept capability depends quite strongly on the scramble policy. There exists a range of raid feint angles which can drastically reduce the number of possible interceptions if a conservative scramble policy is adopted. By being prepared to scramble against an identified raid without close regard to the estimated position of interception the defence can nullify the possible deleterious effects of well-planned indirect routing. Note that we have not investigated the 'campaign' aspects of such a policy. In particular, we assume that abortive fighter sorties do not degrade defences against later raids.

The results presented go some way towards a clarification of the more effective procedures which the defence might adopt in the face of deceptive raid tactics. They indicate the value of ample warning distance to the fighter defences, together with early resolution of raid size in order that sufficient fighters may be scrambled quickly, albeit not necessarily with well-defined interception courses.

Table 2.1

Numerical Inputs to the Study of Raid Indirect Routing

	RAID PARAMETERS	
(A)	Target	108nm along the target axis
(B)	Distance of the feint track from the target, measured perpendicular to the target axis.	270nm
(C)	Angle of Feint	arbitrary
(D)	Track-change point	at the perpendicular to the target
(E)	Raid Speed (constant)	9nm/min
(F)	Raid length - concentrated (point) raid	zero
	GC AND FIGHTER PARAMETERS	
(A)	Warning Distance	(1) 198nm (2) 234nm (3) 270nm (4) 360nm
(B)	Sensor and GC processing and transmission delay at first detection of the raid	5 minutes
(C)	Fighter readiness level	5 minutes
(D)	Scramble Policy: I : scramble if expect to intercept before the target axis II : scramble unless interception of feint track appears impossible III : scramble 20 fighters regardless of their expected interception prospects	

Table 2.1 (Cont'd)

(E)	Delay between take-offs	30 secs (i.e., scramble rate of two per minute)
(F)	Number of fighters available at the base	(1) unlimited (2) twenty
(G)	Fighter speed (constant)	9nm/min
(H)	Delay before fighters, under Close Control (CC) or acting autonomously, react to the raid track-change	zero

Table 2.2

Key to Figures 2.2-2.12

Serial	Definition	Equation
	Scramble Limitations against the feint leg	
A_I	Policy I	(2.5)
A_{II}	Policy II	(2.6)
	Interception Limitations	
	(i) Fighters scrambled to intercept the feint leg	
A_I	Policy I (all fighters scrambled achieve interceptions)	(2.5)
B_{II}	Policy II	(2.11)
	(ii) Fighters scrambled to intercept the second track-leg	
	(a) Policies I and II	
C	Raid detected at least 5 minutes before it changes track	(2.13)
B_{III}	Raid detected between 0 and 5 minutes before it changes track	(2.7)
B'_{III}	Raid detected after it changes track	(2.9)
	(b) Policy III	
B_{III}	Raid detected before it changes track	(2.7)
B'_{III}	Raid detected after it changes track	(2.9)

Notes

(1) Raid track-change at the perpendicular to the target.

(2) Equations (1.5) and (2.14) are used to convert expressions for the maximum acceptable take-off delay into the corresponding expressions giving the number of possible interceptions.

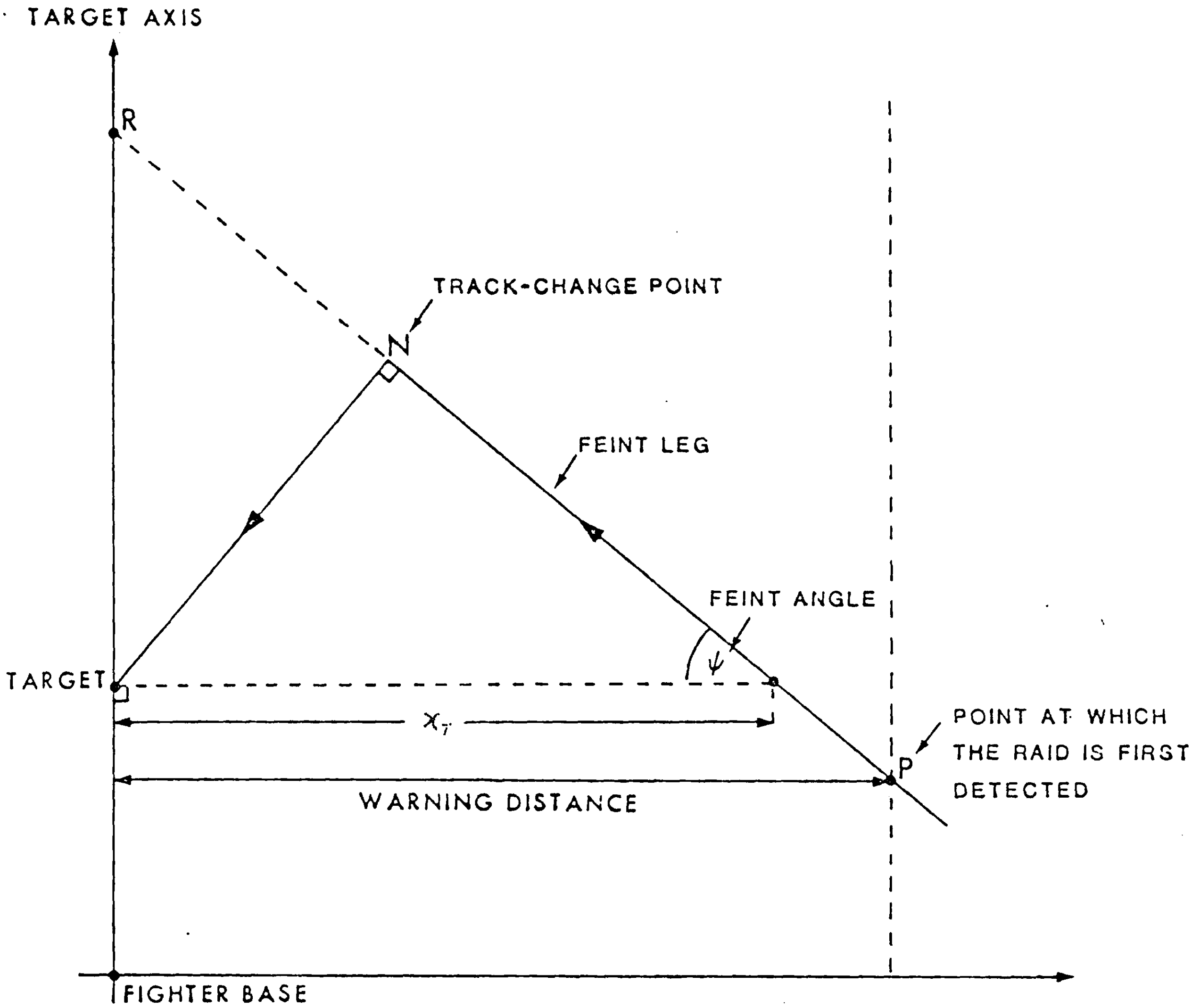


FIGURE 2.1: INDIRECT ROUTING - RAID TRACK PARAMETERS

Note

X_T denotes the distance from the feint track to the target, measured perpendicular to the target axis. The distance X_T and the feint angle ψ together define the feint track.

Warning Distance = 198 nm

SCRAMBLE POLICY I

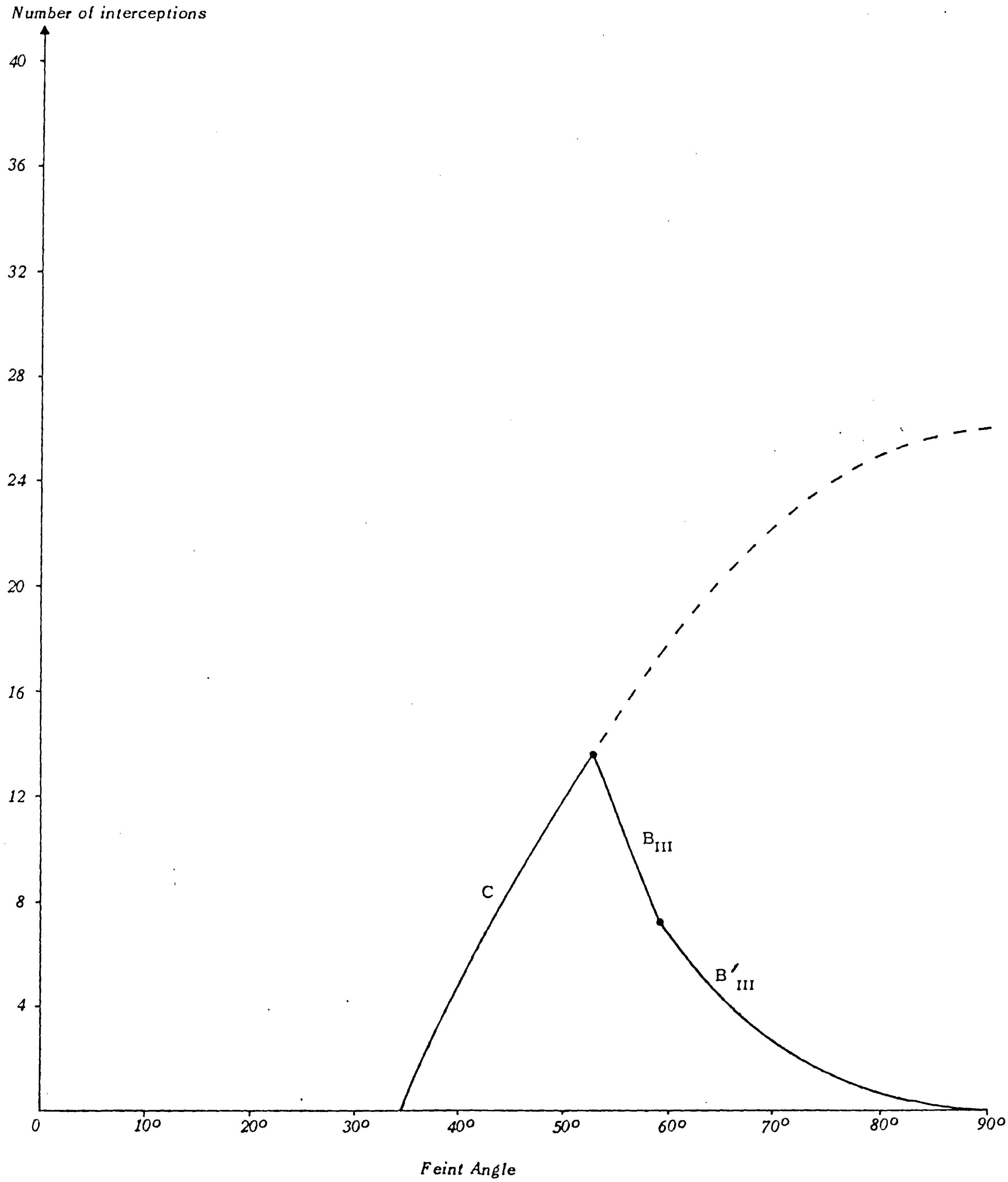


FIGURE 2.2

Warning Distance = 234 nm

SCRAMBLE POLICY I

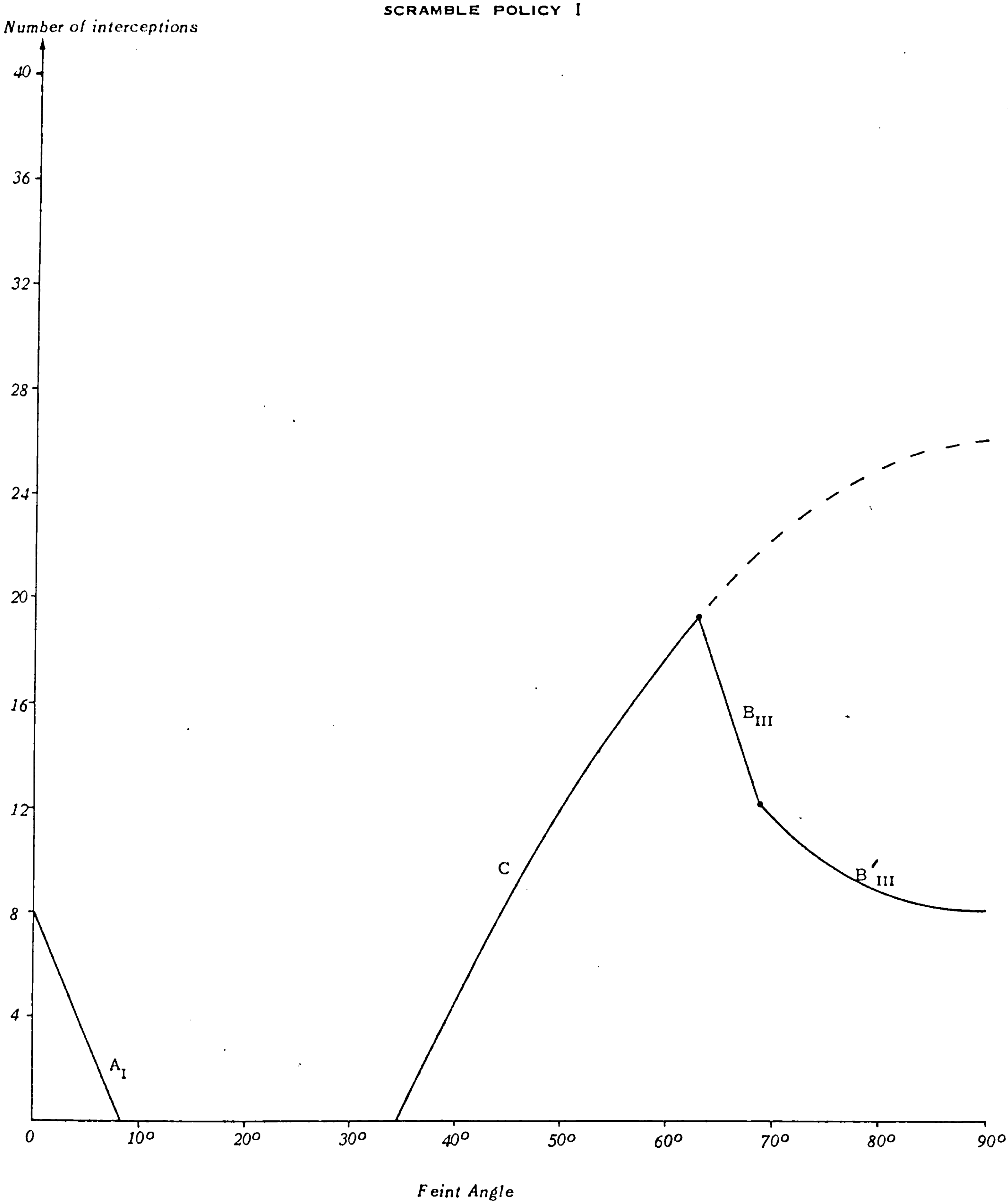


FIGURE 2.3

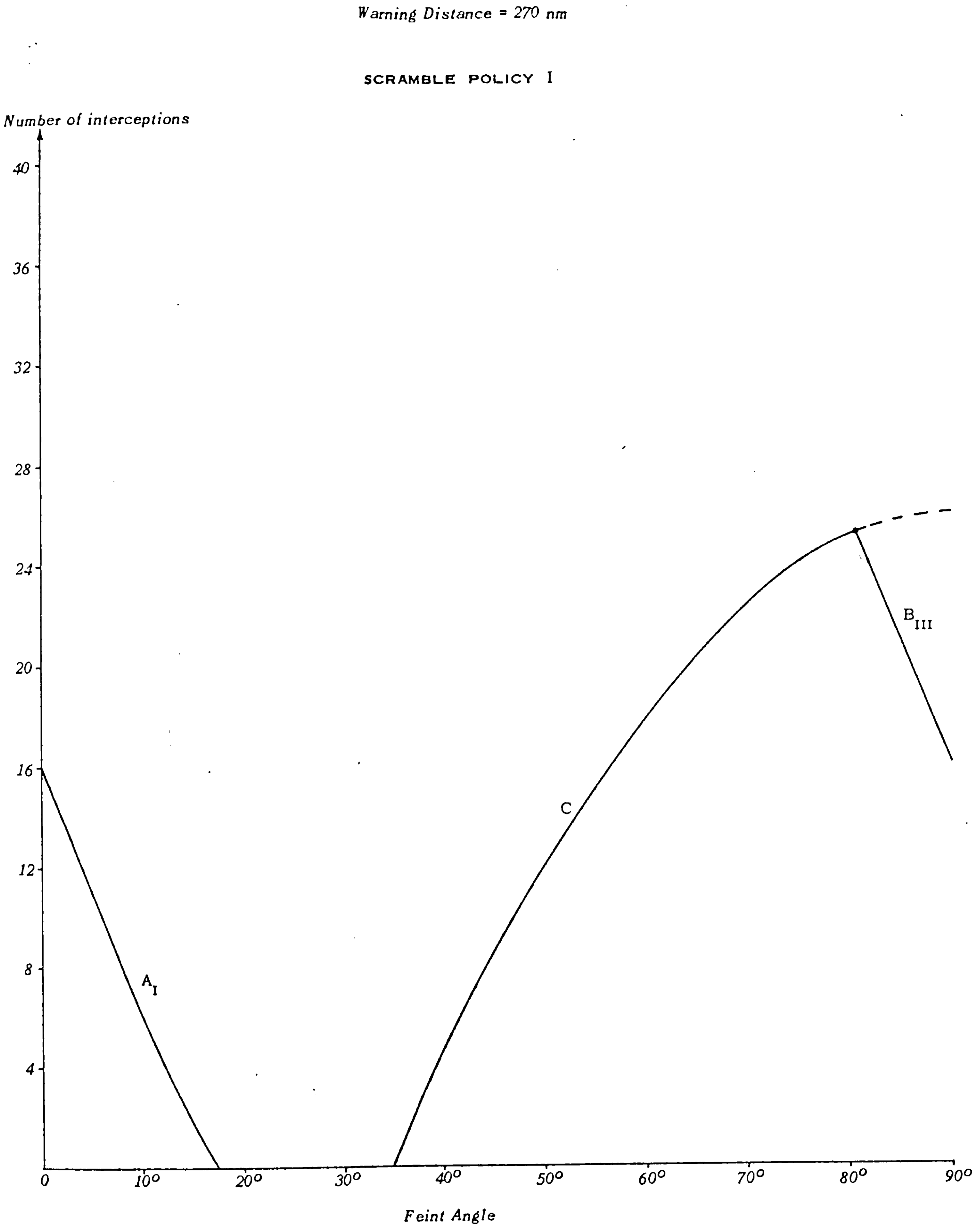

Warning Distance = 270 nm
SCRAMBLE POLICY I
Number of interceptions
40
36
32
28
24
20
16
12
8
4
0
10°
20°
30°
40°
50°
60°
70°
80°
90°
Feint Angle
A_I
C
B_III

FIGURE 2.4

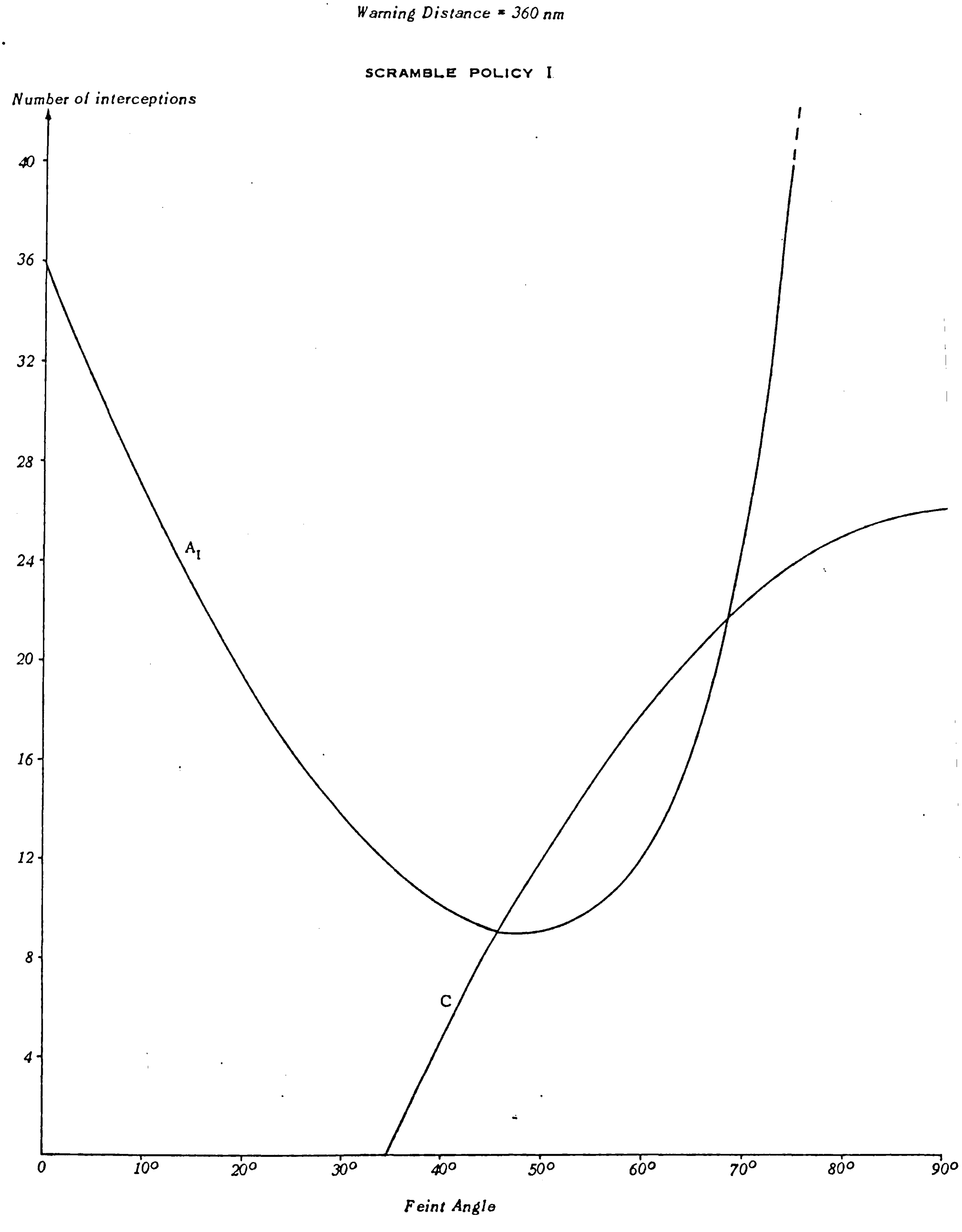

Warning Distance = 360 nm
SCRAMBLE POLICY I
Number of interceptions
40
36
32
28
24
20
16
12
8
4
0
10°
20°
30°
40°
50°
60°
70°
80°
90°
Feint Angle
A_I
C

FIGURE 2.5

Warning Distance = 198 nm

SCRAMBLE POLICY II

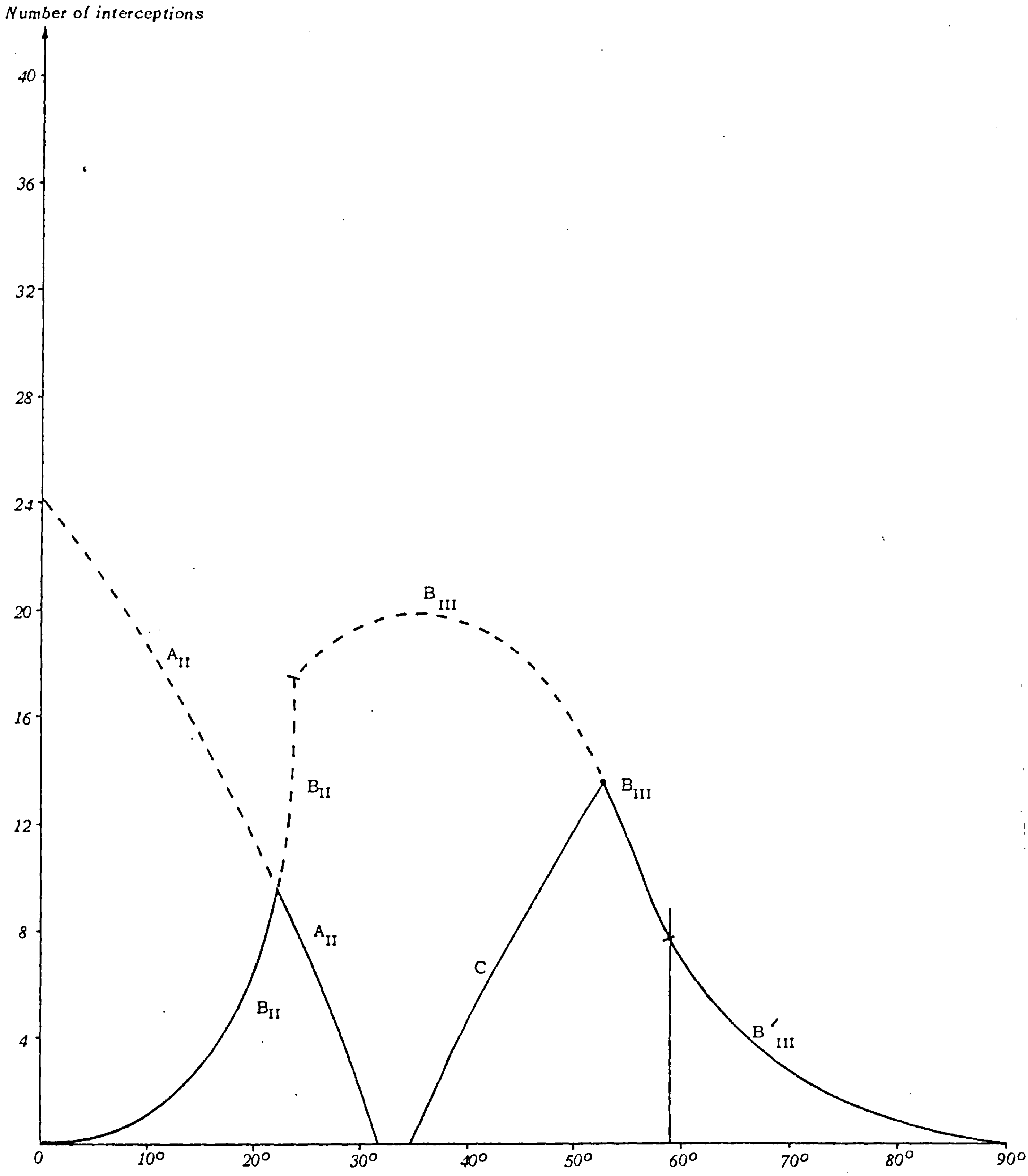


FIGURE 2.6

Warning Distance = 234 nm

SCRAMBLE POLICY II

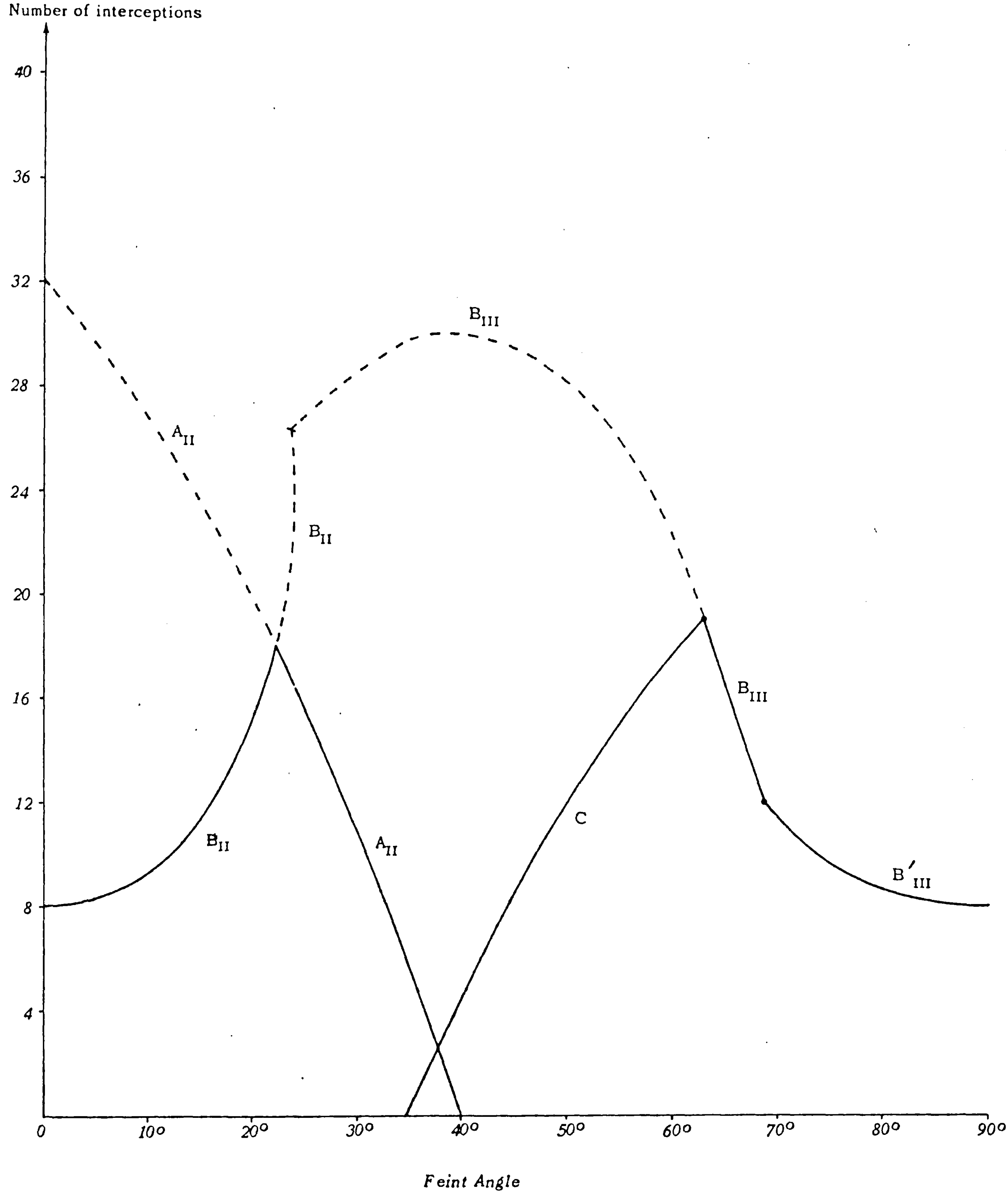


FIGURE 2.7

Warning Distance = 270 nm

SCRAMBLE POLICY III

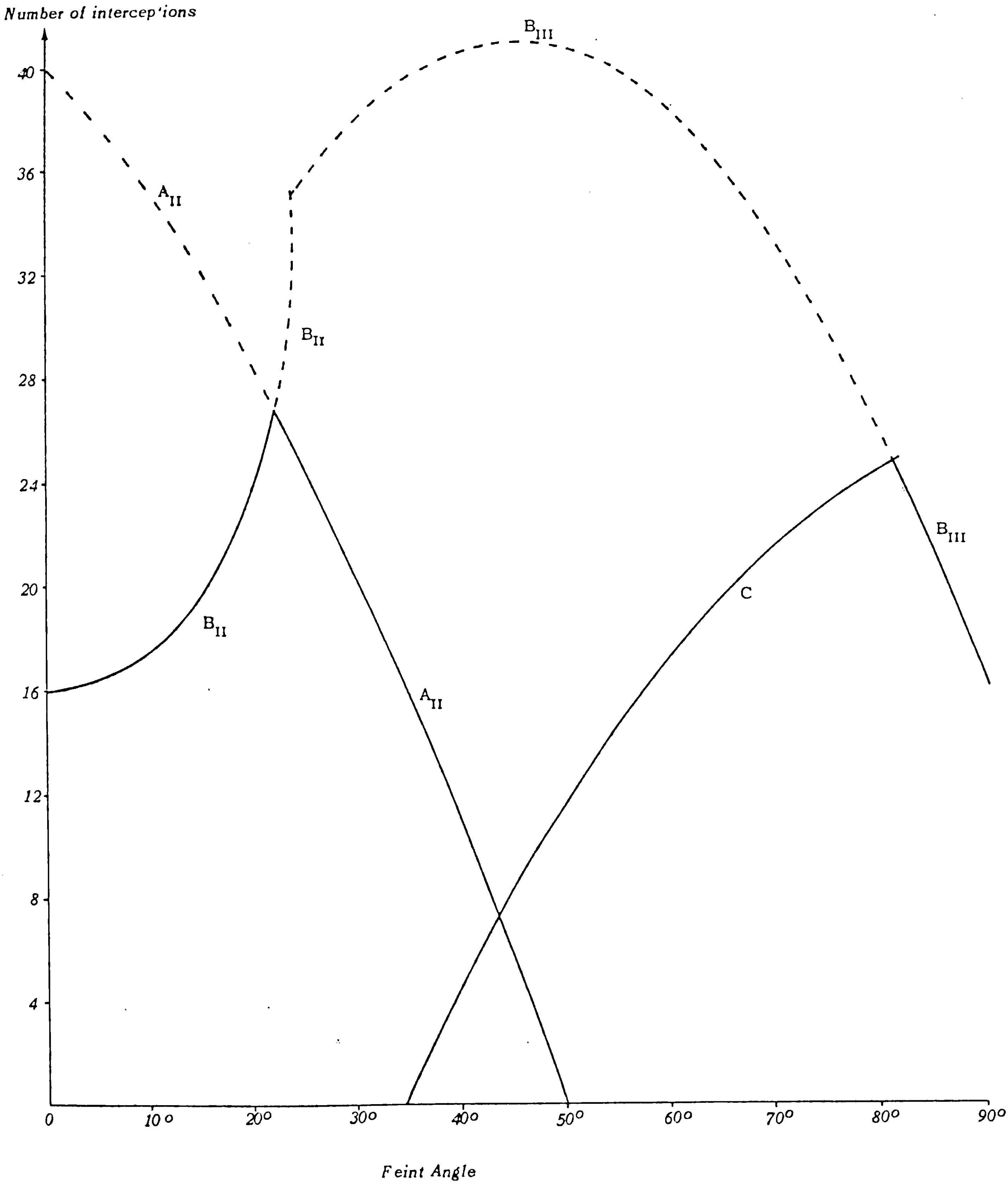


FIGURE 2.8

Warning Distance = 360 nm

SCRAMBLE POLICY II

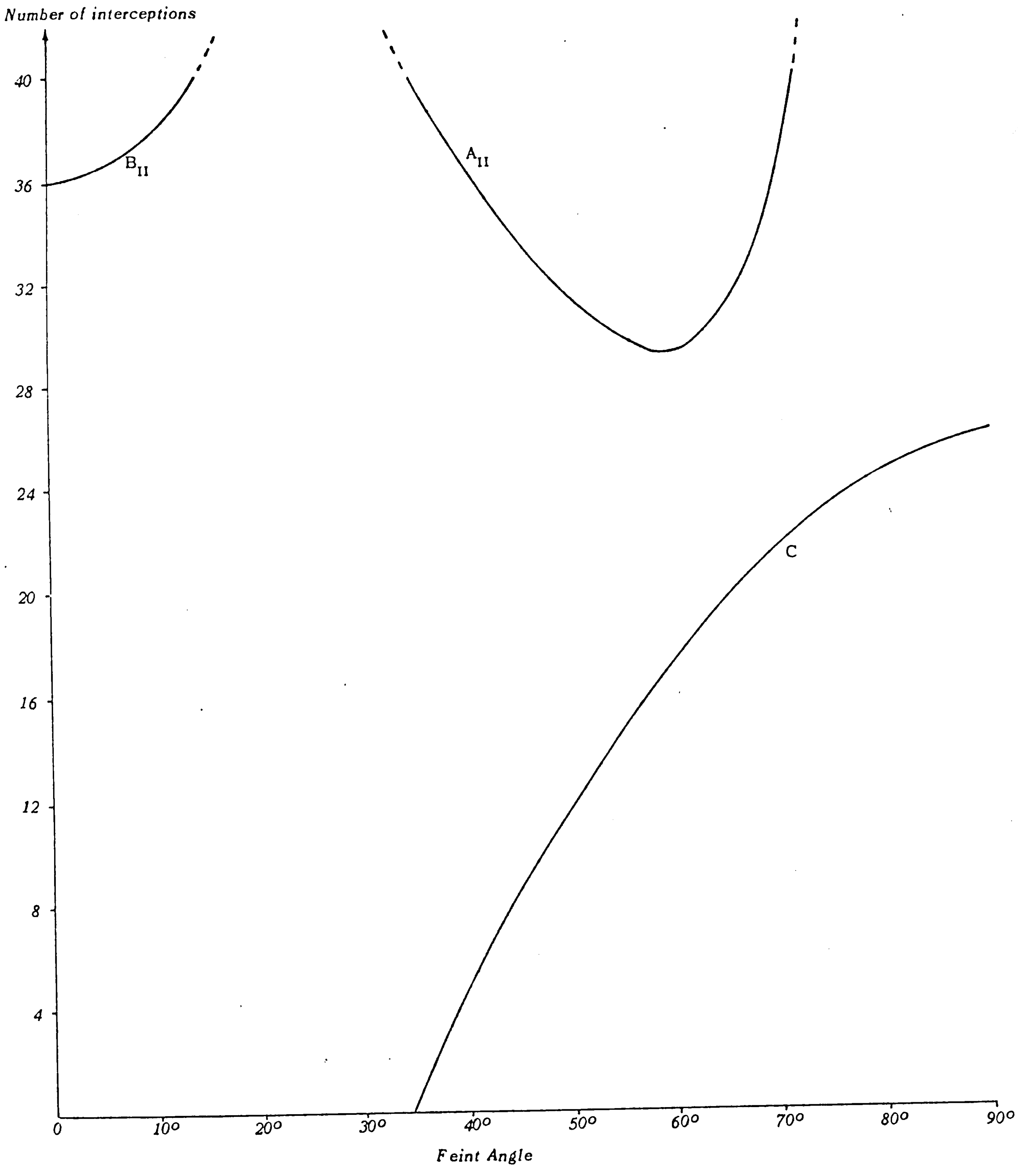


FIGURE 2.9

Warning Distance = 198 nm

SCRAMBLE POLICY III

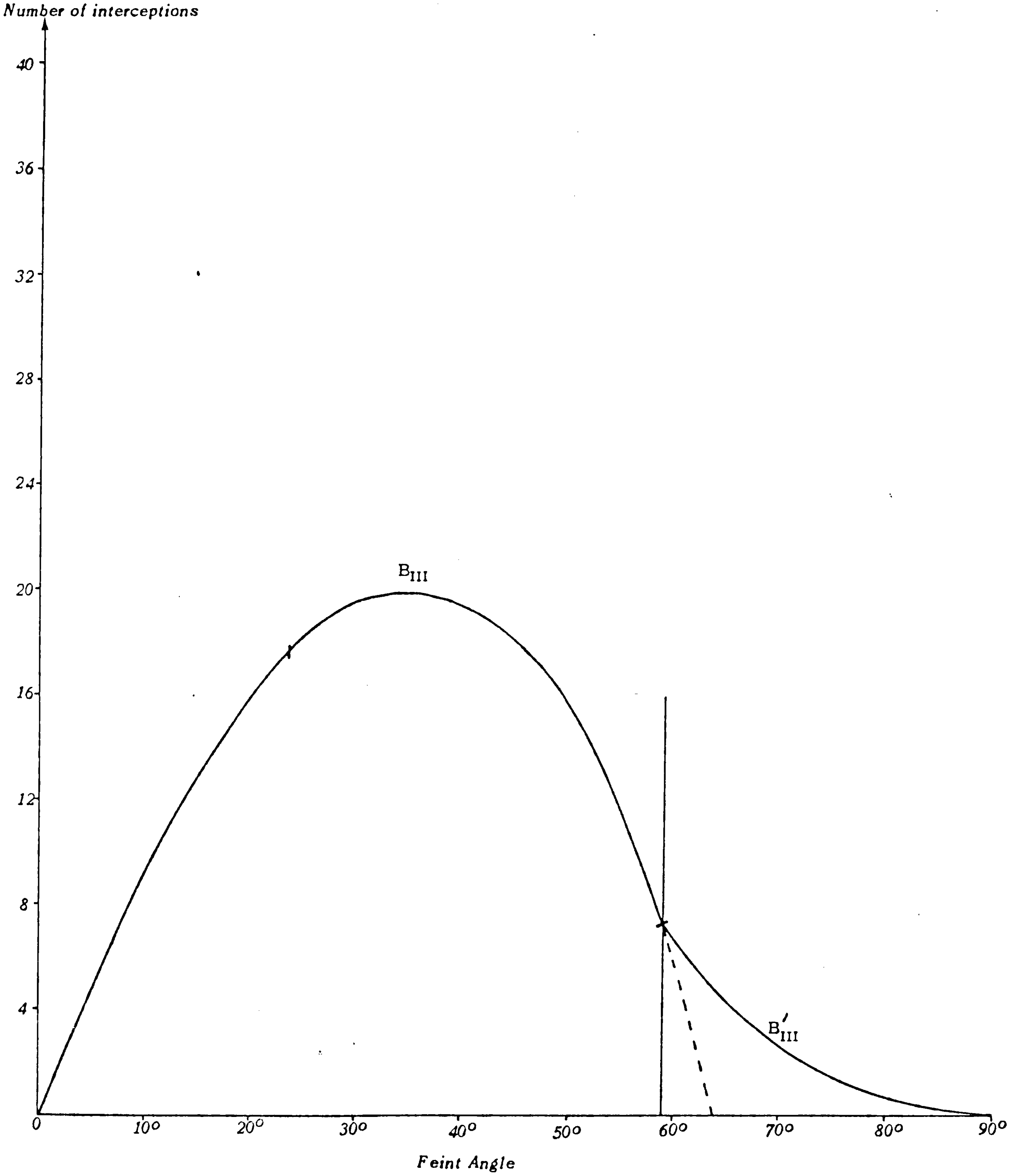
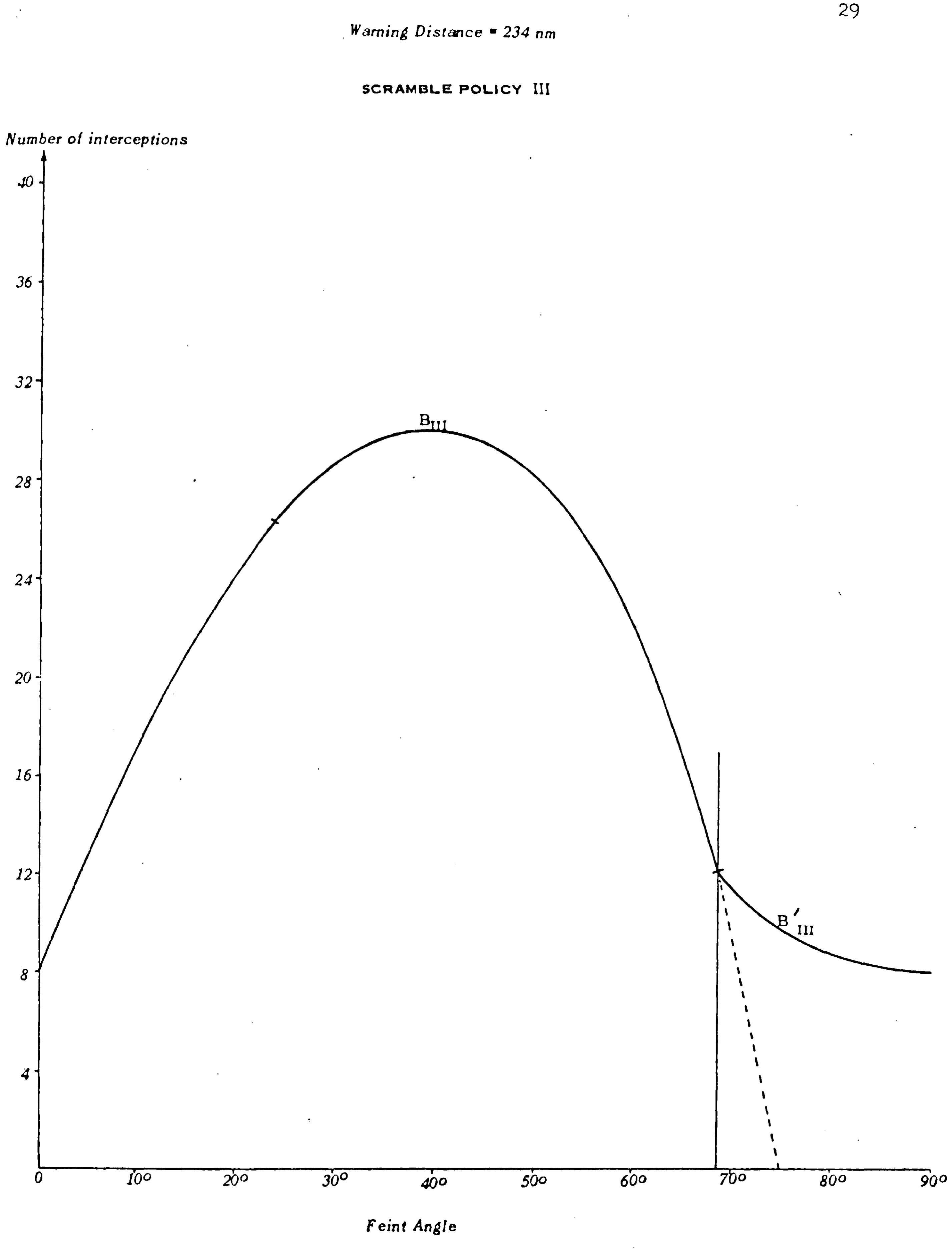


FIGURE 2.10



Warning Distance = 234 nm
SCRAMBLE POLICY III
Number of interceptions
40
36
32
28
24
20
16
12
8
4
B_III
B'_III
0
10°
20°
30°
40°
50°
60°
70°
80°
90°
Feint Angle

FIGURE 2.11

Warning Distance = 270 nm and 360 nm

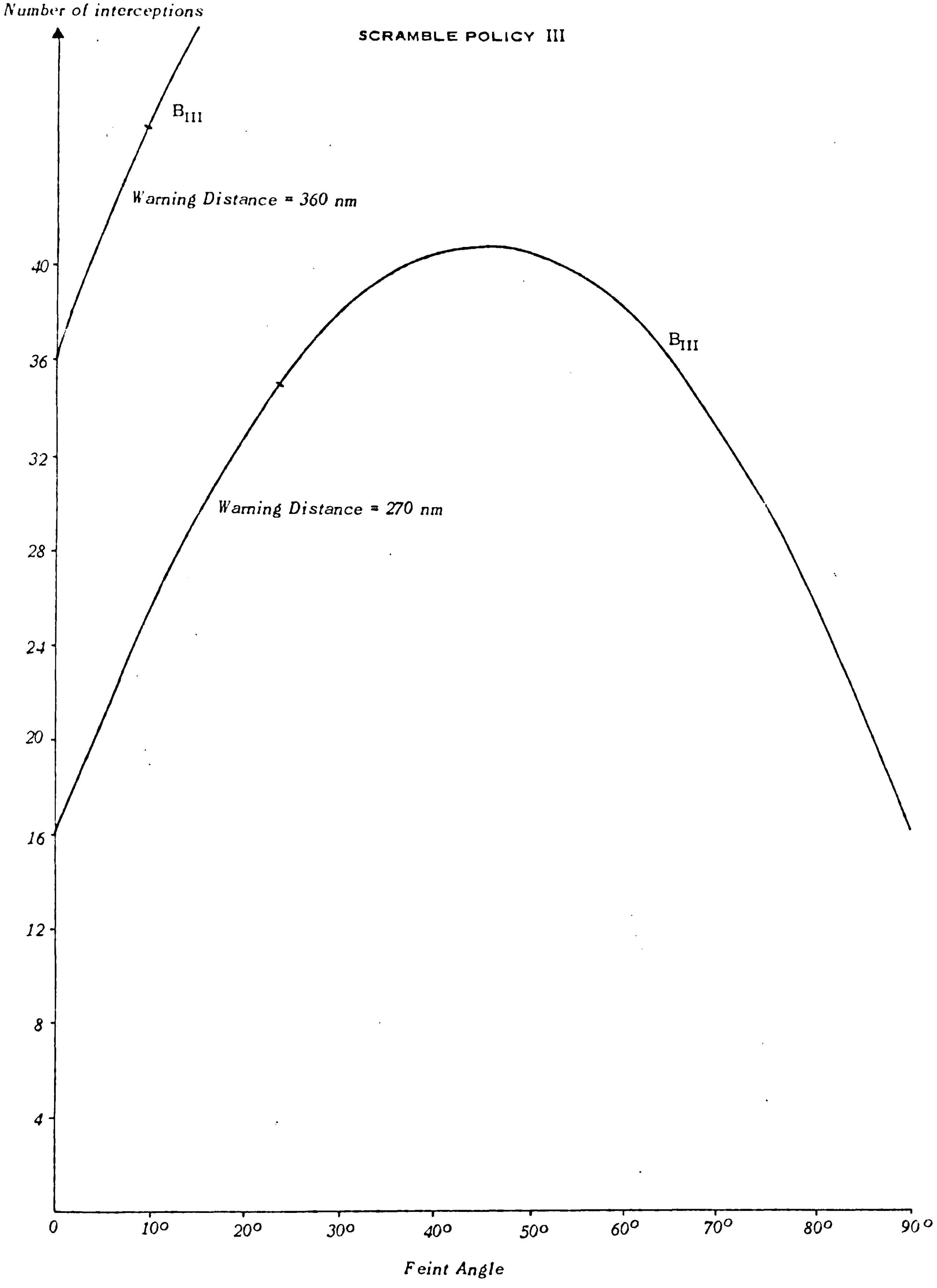
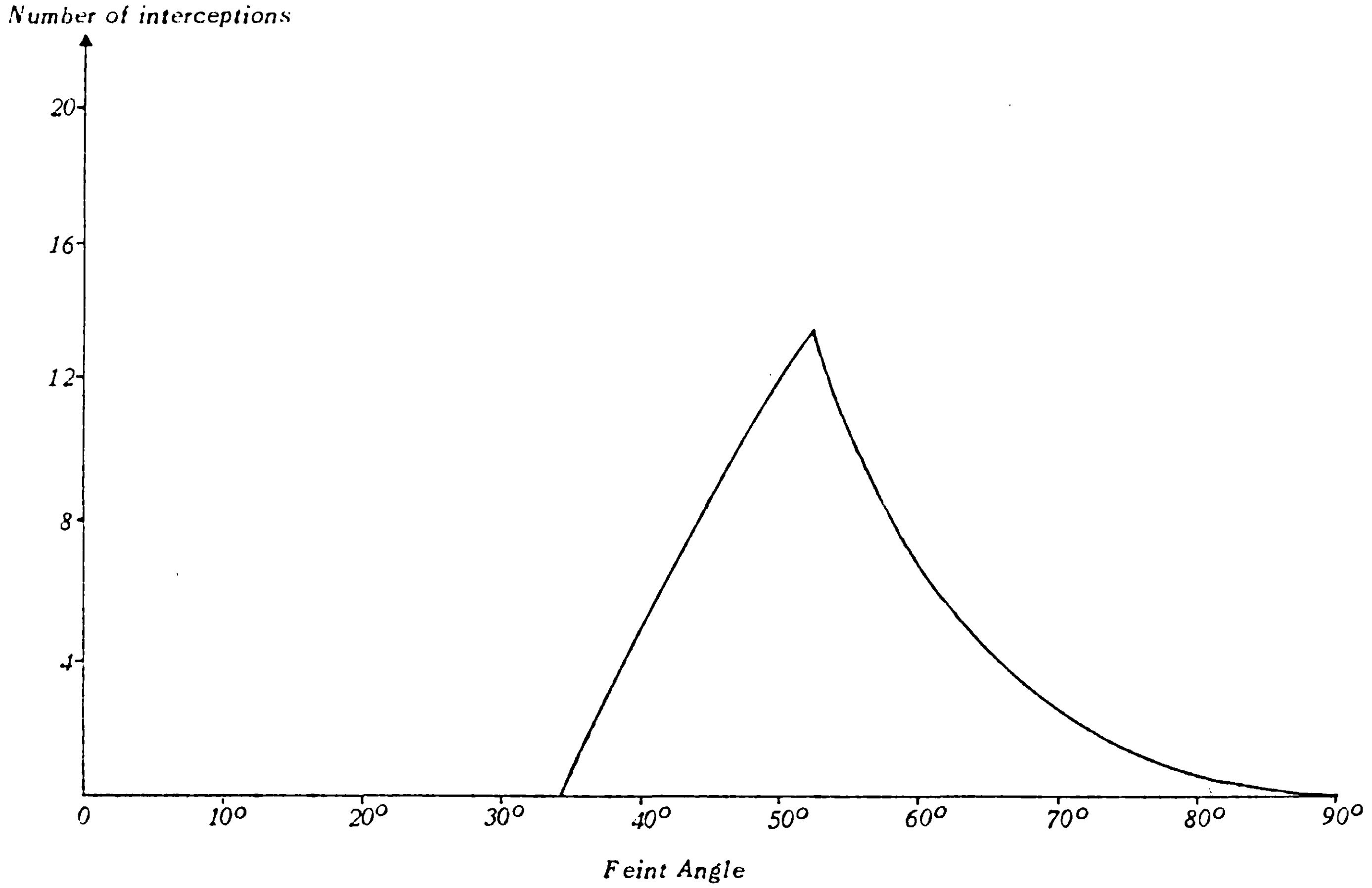


FIGURE 2.12

Warning Distance = 198nm; Scramble Policy I



Warning Distance = 198nm; Scramble Policies II and III

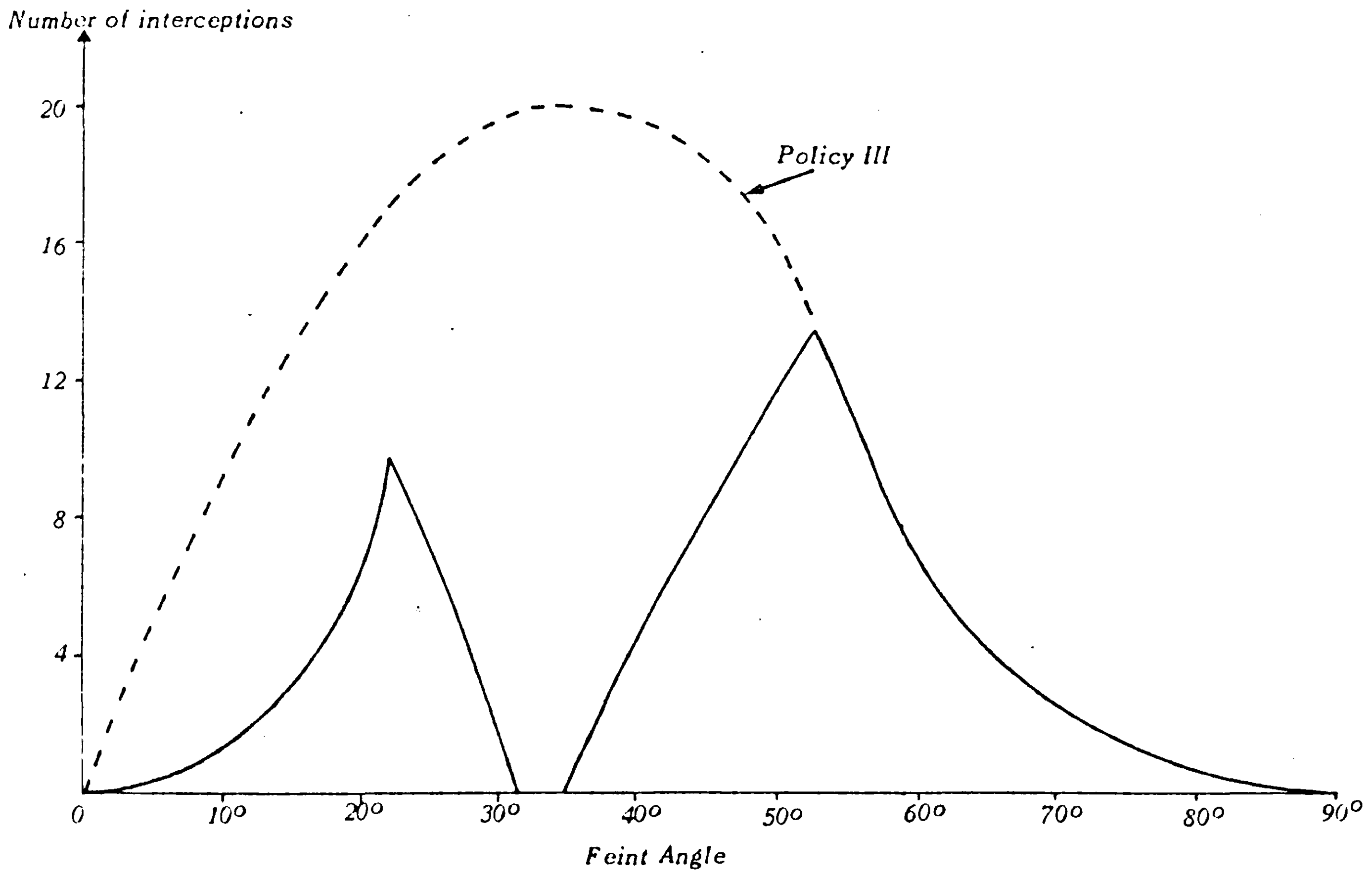
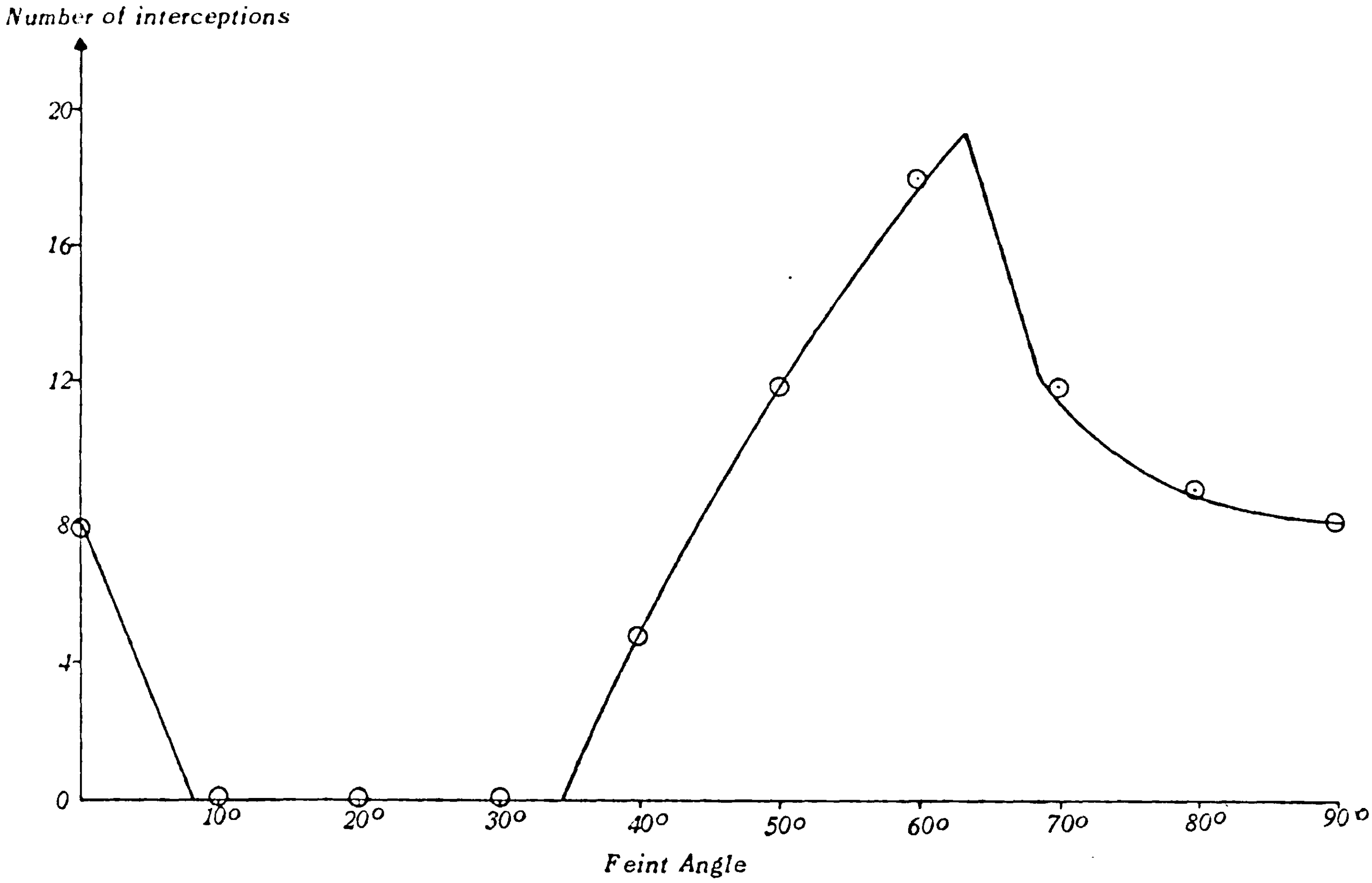


FIGURE 2.13

Warning Distance = 234nm; Scramble Policy I



Warning Distance = 234nm; Scramble Policies II and III

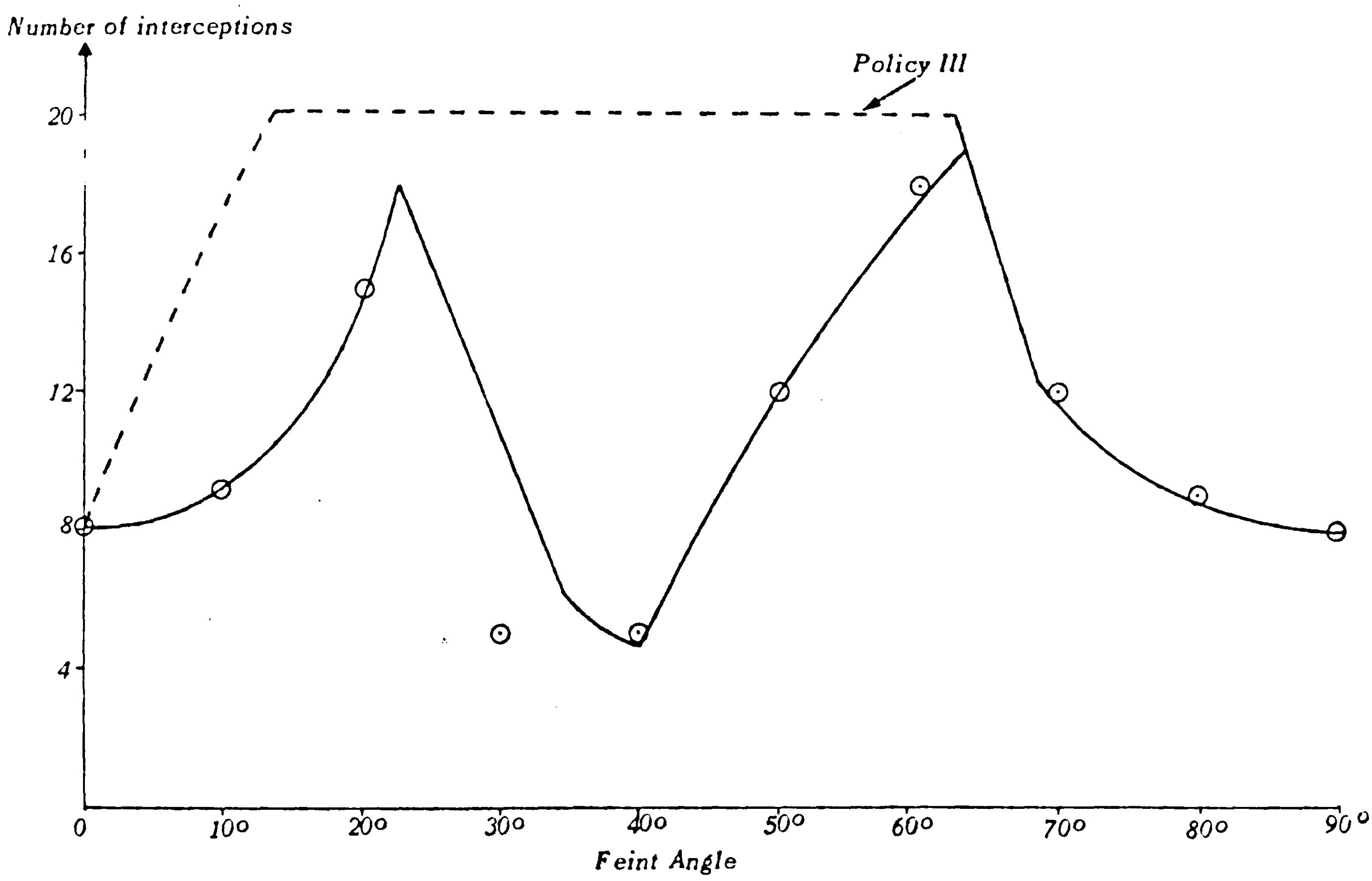
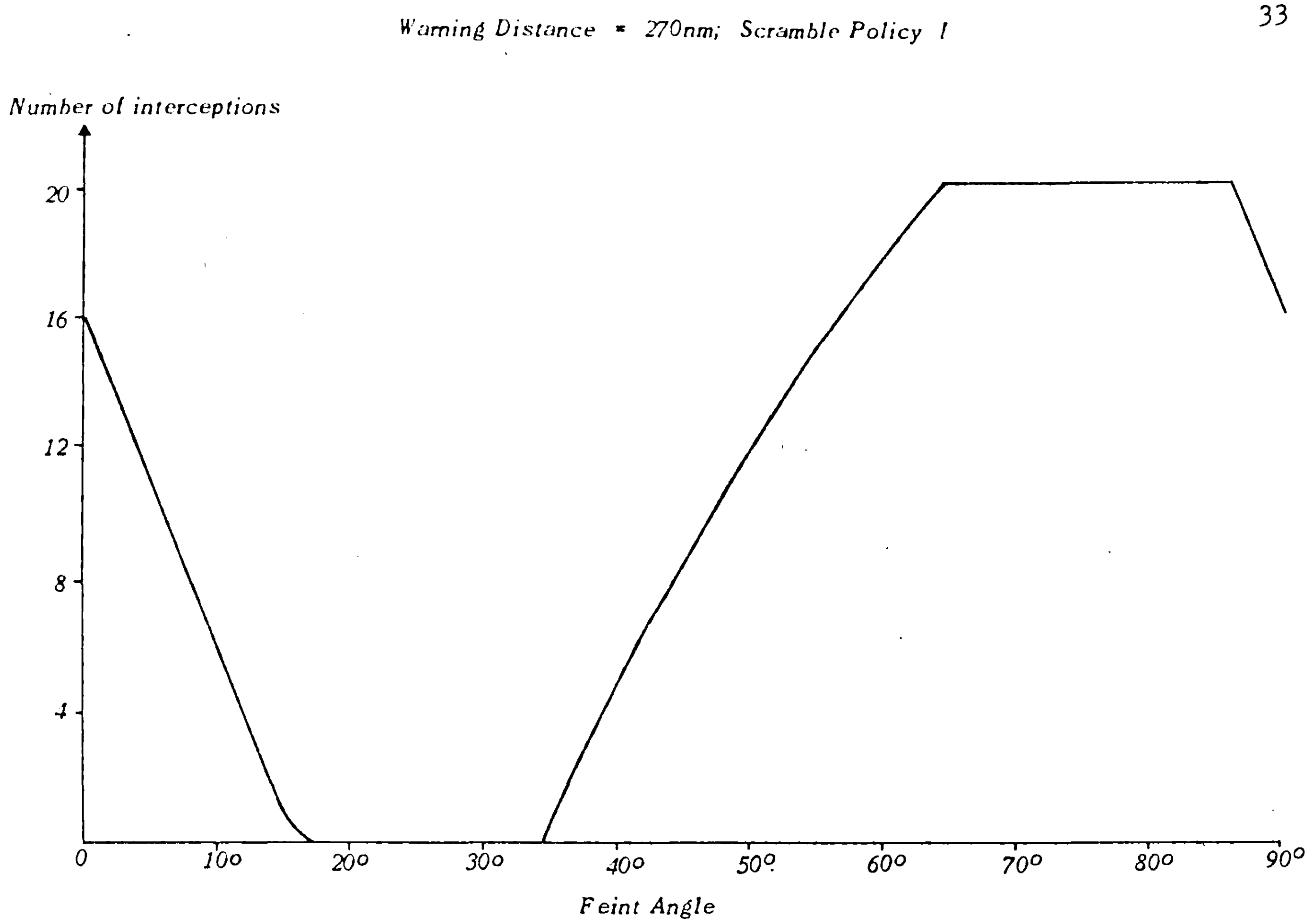


FIGURE 2.14


Warning Distance = 270nm; Scramble Policy I
Number of interceptions
20
16
12
8
4
0
10°
20°
30°
40°
50°
60°
70°
80°
90°
Feint Angle

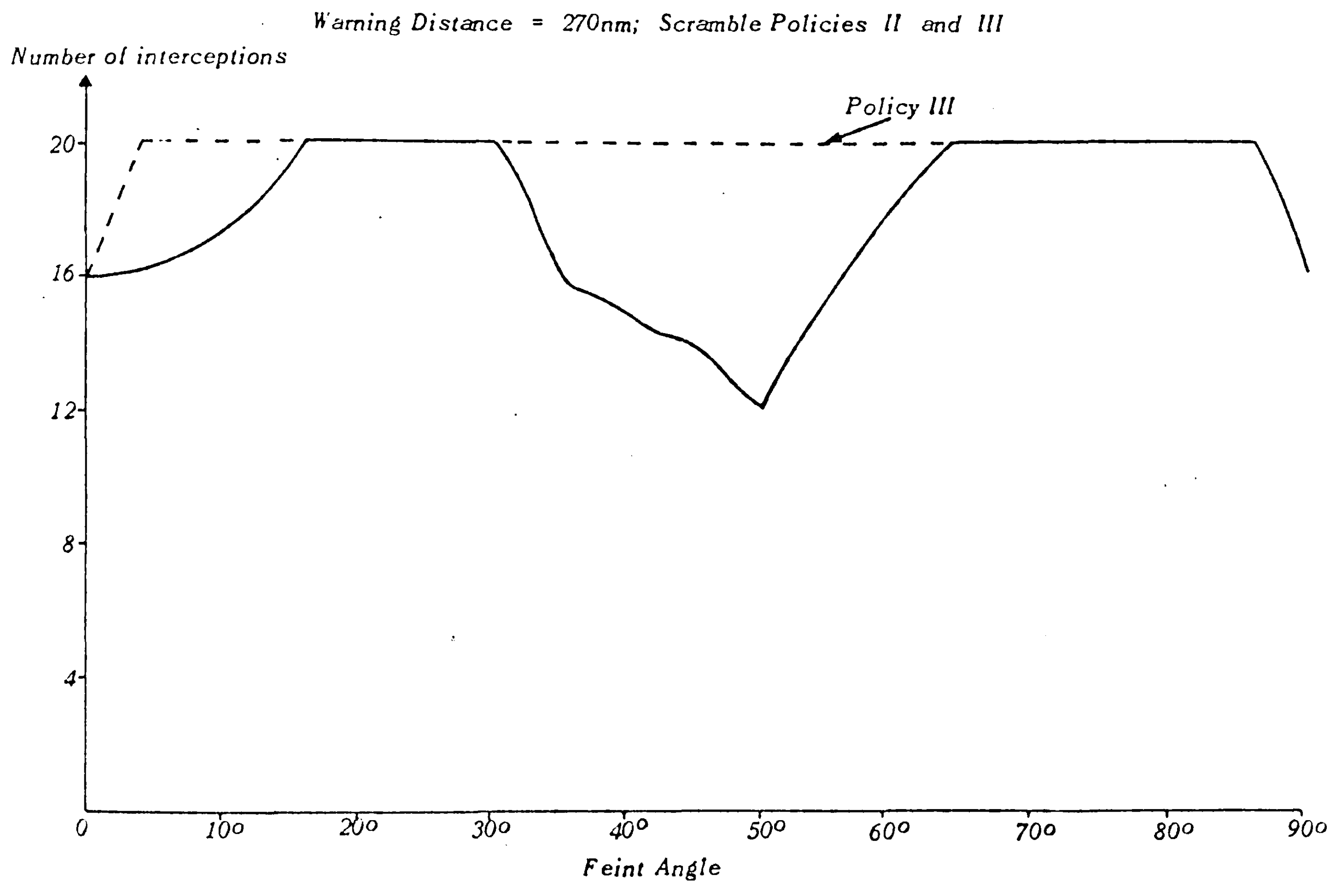
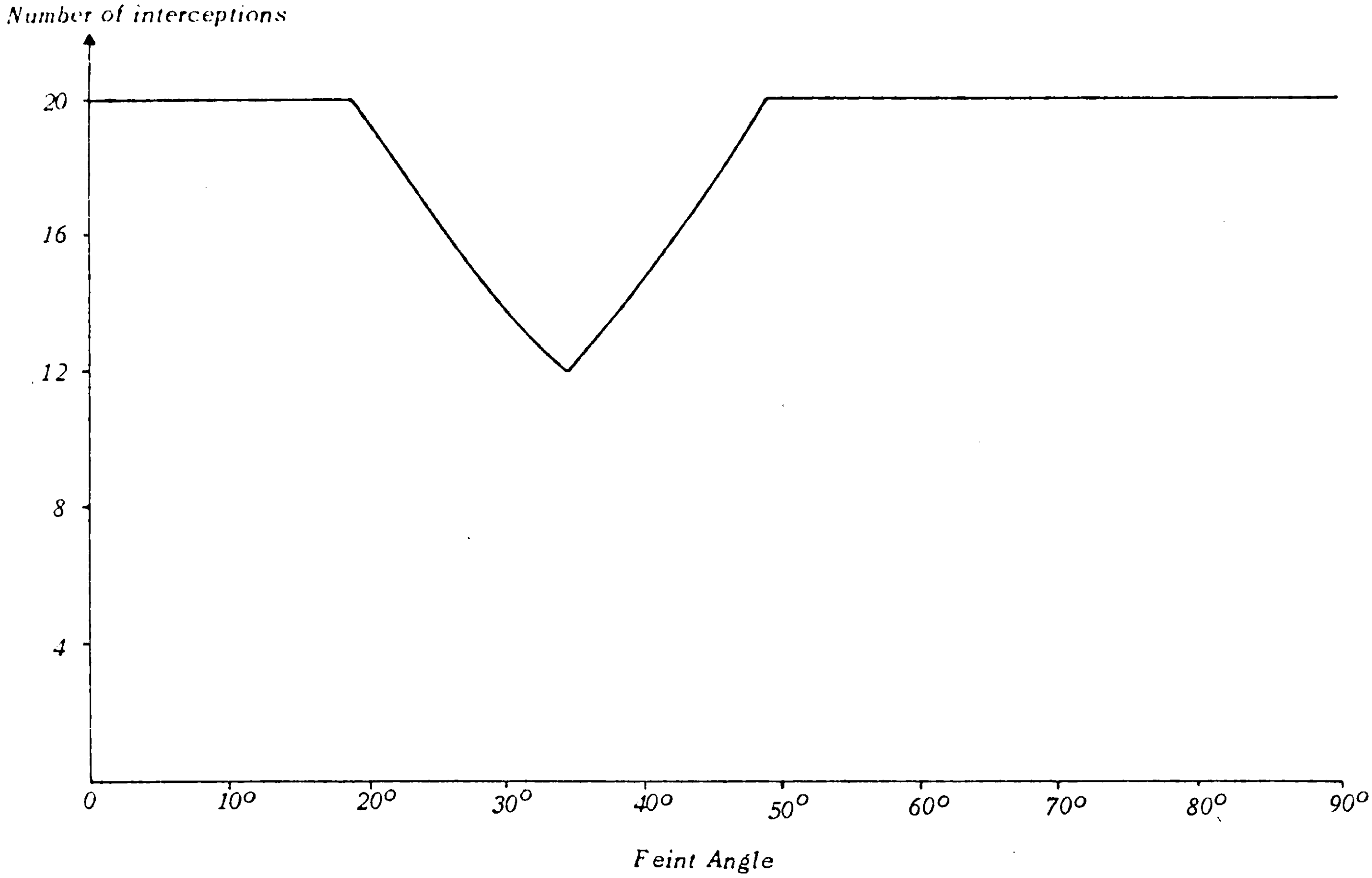

Warning Distance = 270nm; Scramble Policies II and III
Number of interceptions
Policy III
20
16
12
8
4
0
10°
20°
30°
40°
50°
60°
70°
80°
90°
Feint Angle

FIGURE 2.15

Warning Distance = 360nm; Scramble Policy I



Warning Distance = 360nm; Scramble Policies II and III

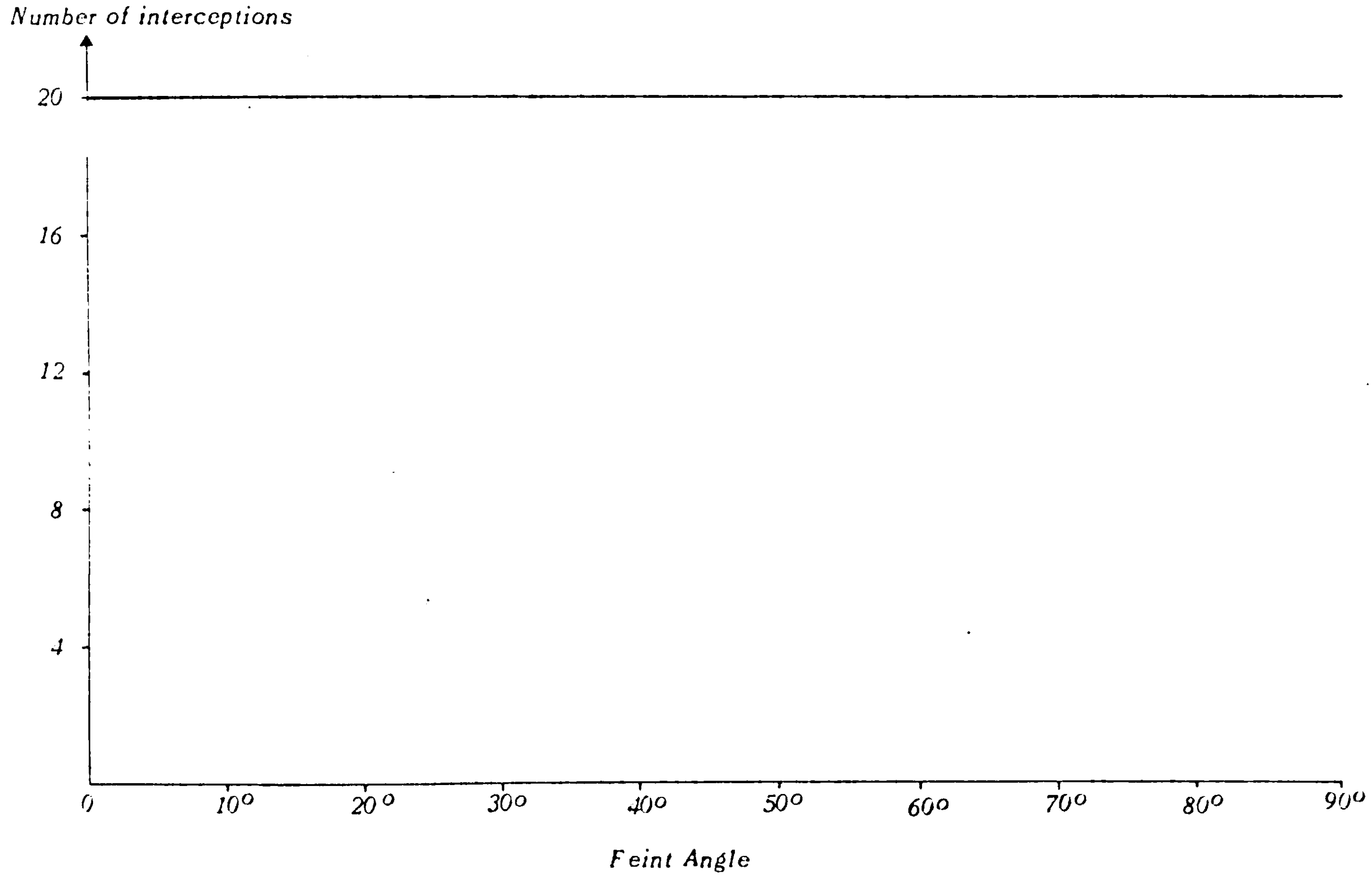


FIGURE NO. 2.16

APPENDIX A

FIGHTER SCRAMBLE POLICIES

Introduction

To begin the analysis of fighter interception capability it is necessary to specify the conditions under which fighters are scrambled. Four scramble policies with associated control procedures are considered.

I Scramble is continued as long as interception is expected to occur before or over the target axis; the initial fighter track is directed towards the estimated interception point.

II Scramble is continued until fighters cease to have the potential to intercept the raid if it continues along its estimated track, i.e., as long as fighters possess a geometrically feasible interception course. In practice, the implementation of such a policy must sometimes imply a degree of doubt by the fighter controller that the target lies along the feint track, since fuel limitations are not taken into account when the fighter is scrambled against the feint track. Again, the initial fighter track is directed towards the expected interception point.

III Under policy III the feint leg is ignored and fighters fly along the target axis to an ideally suited combat air patrol position, directly over the target. This policy is included as an analytical control to illustrate the best that can be achieved by the fighter defences; it differs from the others in that it presupposes Ground Control has some knowledge of the raid target.

IV Policy IV is a standard fuel-limitation policy, in which interception must be expected within a specified time or distance after take-off for scramble to proceed.

Although Figure 1.2 only is used to illustrate the raid and fighter track geometry, the results in this Appendix apply to all of the configurations given by cases (1)-(6). Throughout, D denotes the maximum acceptable take-off delay under each scramble policy. Before we derive D for policies I-IV it is convenient to calculate the expected time to interception after scramble and the heading, ϕ, which defines the initial fighter interception course.

Time to Interception

From Figure A1, the raid is at the point R , with coordinates (x_1, y_1) given by (1.6), at the moment the fighter begins its interception course. Its corresponding polar coordinates are r_1 and θ_1. If P denotes the expected interception point with coordinates (X,Y) we have

$$X = x_1 - UT\cos\psi = VT\cos\phi$$

and (A1)

$$Y = y_1 + UT\sin\psi = VT\sin\phi \ ,$$

where T is the expected time to interception and ϕ denotes the fighter heading. Equation (A1) gives

$$T^2 \ (U^2-V^2) - 2r_1\cos\omega.UT + r_1^{\ 2} = 0 \ , \qquad \text{(A2)}$$

where, from Figure A1,

$$\omega = \theta_1 + \psi \qquad \text{(A3)}$$

and

$$r_1^{\ 2} = x_1^{\ 2} + y_1^{\ 2} \ ,$$

$$\theta_1 = \tan^{-1}(y_1/x_1) \ . \qquad \text{(A4)}$$

If U=V then equation (A2) has solution

$$T = \frac{r_1}{2U\cos\omega} \ , \qquad (U=V) \qquad \text{(A5)}$$

while, if $U \neq V$, taking the minimum time interception we have

$$T = \frac{r_1}{U^2-V^2} \ (U\cos\omega - (V^2-U^2\sin^2\omega)^{\frac{1}{2}}) \ . \qquad (U \neq V) \qquad \text{(A6)}$$

(To be more exact, if $V > U$ there is just one valid solution to (A2), given by (A6). If $V < U$, there are two real positive roots of (A2), provided that $V \geq U\sin\omega$, and the smaller value is given by (A6).)

Fighter Heading

Given the expected time to interception, T, from equation (A5) or (A6), the fighter heading ϕ is immediately available from equation (A1)

$$\cos\phi = \frac{x_o - U(D+T)\cos\psi}{VT}, \qquad \sin\phi = \frac{y_o + U(D+T)\sin\psi}{VT}. \tag{A7}$$

A useful alternative expression for ϕ may be found as follows. From Figure A1 we may write

$$\phi = \gamma + \theta_1, \tag{A8}$$

where

$$\sin\gamma = \frac{p}{r_1} \cdot \frac{U}{V} \tag{A9}$$

and p is given by equation (1.7). Substituting into (A8) gives

$$\sin\phi = \left\{ x_1 . p . \frac{U}{V} + y_1 \left(r_1^2 - p^2 . \frac{U^2}{V^2} \right)^{1/2} \right\} / r_1^2 \tag{A10}$$

with a similar expression for $\cos\phi$.

The above expressions hold for each of the configurations (1)-(6) shown in Figure 1.3, provided that the angles γ, ϕ and θ_1 are interpreted in the following sense:

γ is the angle from OR_1 to OP, measured anticlockwise;
ϕ is the angle from the positive x-axis to OP, measured anticlockwise;
θ_1 is the angle from the positive x-axis to OR_1, measured such that

$$-\pi < \theta_1 \leq \pi .$$

Scramble Policy I

From Figure 1.2 the maximum take-off delay D for scramble under this policy, is such that

$$\frac{/OF/}{V} = \frac{/RF/}{U} \quad . \tag{A11}$$

This gives

$$\frac{/y_T + y_M/}{V} = \frac{x_o \sec\psi - UD}{U} \quad ,$$

so that

$$D = \frac{1}{U} \left(x_o \sec\psi - \frac{U}{V} (/y_T + y_M/)\right) \quad . \tag{A12}$$

Scramble Policy II

From equation (A9) and Figure A1, the limiting case for interception under this policy is given by

$$/p/ . \frac{U}{V} = r_1 . \qquad (A13)$$

Substituting for p and r_1 from (A4) and (1.7) into (A13) leads to

$$x_1 = p(\sin\psi \pm k.\cos\psi)$$

and

$$y_1 = p(\cos\psi \mp k.\sin\psi) , \qquad (A14)$$

where

$$k^2 = \frac{U^2}{V^2} - 1 . \qquad (A15)$$

Hence from (1.6) D is given by

$$D = \frac{1}{U\cos\psi} (x_o - p(\sin\psi \pm k.\cos\psi)) . \qquad (A16)$$

It may be seen that in the above equations the top sign is taken except in cases (3) and (6). Note that from (A15), if V>U there is no limiting point for fighter scramble under this policy.

Finally, from Figure 1.3 it is possible in cases (1), (5) and (6) that the maximum take-off delay D for scramble under this policy is such that the limiting interception occurs before the raid reaches the target axis. It follows easily from the above analysis that the coordinates (X_e, Y_e) of the limiting interception point P_e under scramble policy II are given by

$$X_e = -\frac{y_1}{k} , \quad Y_e = \frac{x_1}{k} , \qquad \text{cases (1), (2), (4), (5)} \qquad (A17)$$

and

$$X_e = \frac{y_1}{k} , \quad Y_e = -\frac{x_1}{k} . \qquad \text{cases (3), (6)} \qquad (A18)$$

By substituting the appropriate expression for y_1 from equation (A14) into the above expressions for X_e we find that, in cases (1),(5) and (6), we do not need to consider scramble policy I when

$$U > V$$

and

$$/\tan\psi/ > \frac{1}{k} . \qquad (A19)$$

Scramble Policy III

From Figure 1.2, under this policy the maximum delay D before take-off for successful interception is found from

$$\frac{y_T}{V} = \frac{/RC/ + /CT/}{U} \quad , \qquad (A20)$$

where C is the point at which the raid abandons its feint track and heads directly for the target. We note the following expressions for /RC/ and /CT/ , which hold for cases (1)-(6):

$$/CT/ = q/\mathrm{cosec}\Theta/ \; ,$$

$$/RC/ = x_o \sec\psi - UD - /CF/ \; , \qquad (A21)$$

and

$$/CF/ = q/\cot\Theta + \tan\psi/ \; ,$$

where q is given by (1.8). Substituting these into equation (A20) gives

$$D = \frac{1}{U}\left(x_o \sec\psi + q(/\mathrm{cosec}\Theta/ - /\cot\Theta + \tan\psi/) - \frac{U}{V}\cdot y_T\right). \qquad (A22)$$

Equation (A22) does not hold if the raid is not detected by Ground Control sensors until after its track-change. In this case, from Figure 1.2, D is given for all six cases by

$$\frac{y_T}{V} = \frac{x_o \,\mathrm{cosec}(/\psi/-/\theta/) - UD}{U} \quad , \qquad (A23)$$

i.e.

$$D = \frac{1}{U}\left(x_o \mathrm{cosec}(\,/\psi/ - /\theta/\,) - \frac{U}{V}\cdot y_T\right). \qquad (A24)$$

Scramble Policy IV

Under this policy fighters are not scrambled if they do not have sufficient fuel to carry out the expected interception; this is therefore the most immediately practical of the four scramble policies considered here. It may be regarded as a policy under which a fighter is only scrambled if it expects to intercept within a specified time, say $\bar{T}$, after take-off. Hence if T is the expected time to interception, given by (A5) and (A6), this policy may be expressed simply as

$$T < \bar{T} \; . \qquad (A25)$$

In the limiting case when the fighter achieves an interception just as it reaches the limit of its endurance, so that $T = \bar{T}$, then by rearranging equation (A2) we have

$$(r_1 \cos\omega)^2 - 2U\bar{T}\,(r_1 \cos\omega) + (\bar{T}^2(U^2-V^2)+p^2) = 0 \; . \qquad (A26)$$

This gives

$$r_1 \cos\omega = U\bar{T} - ((V\bar{T})^2-p^2)^{\frac{1}{2}} \; . \qquad (A27)$$

Now from Figure A1 we may write

$$r_1 \cos\omega = r_o \cos\omega_o - UD \; ,$$

where

$$r_o \cos\omega_o = x_o \sec\psi - p.\tan\psi \; . \qquad (A28)$$

Substituting into (A27) gives

$$D = \frac{1}{U}\,(x_o \sec\psi - p.\tan\psi - U\bar{T} + ((V\bar{T})^2 - p^2)^{\frac{1}{2}}) \; . \qquad (A29)$$

(To be more exact, taking the limiting case $T = \bar{T}$ as in (A26) - (A29) is only meaningful if $r_1(U\cos\omega)/(U^2-V^2) > \bar{T}$ (see (A6)), for otherwise scramble always occurs.)

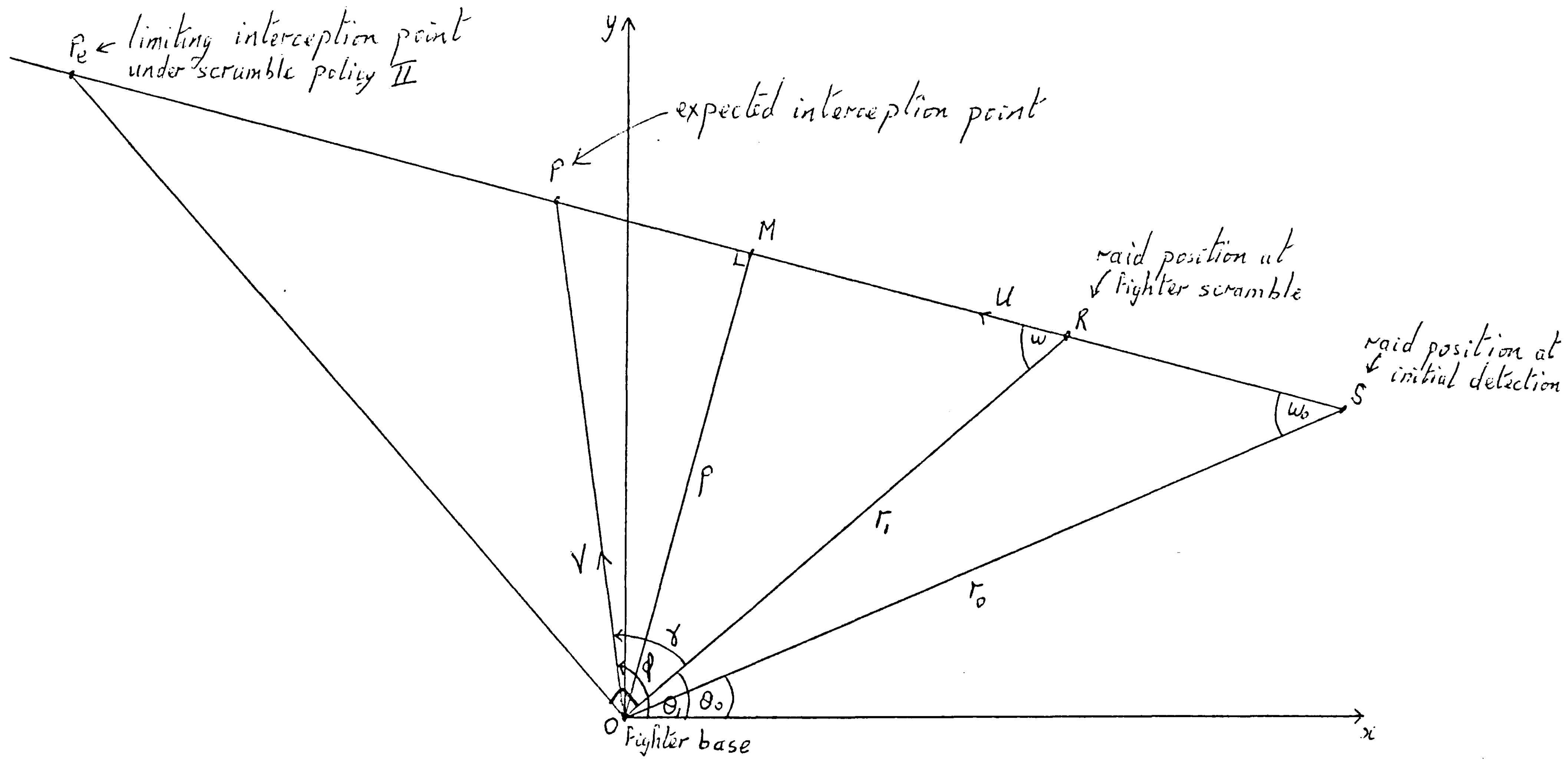


Figure A1. Expected Interception Point

APPENDIX B

CATEGORIES OF INTERCEPTION

Introduction

The analysis is considerably more tractable if interceptions are categorised into the following three distinct types, illustrated in Figure B1:

(i) interceptions on the feint leg;

(ii) interceptions on the second track-leg track by fighters reallocated from expected interceptions on the feint leg;

(iii) interceptions on the second track-leg by fighters scrambled to intercept the second track-leg. This comprises all interceptions under scramble policy III, while under the other policies it corresponds to fighters scrambled to intercept after detection of the track-change by the warning sensors.

We consider each of the interception categories (i)-(iii) in turn. The maximum allowable take-off delay for interception in categories (i) and (iii) is easily found. The criteria for interceptions under category (ii) are more involved and correspondingly more interesting. It is instructive to specify the track-change point C not only by the track-change angle Θ (see Figure 1.2) but also by the time t_s at which the track-change occurs after the raid is first detected at the point S. For practical considerations we may assume that the track-change occurs before the raid reaches the target axis, so that from Figure 1.2 the parameters Θ and t_s are related by

$$/SC/ = Ut_s = /SF/ - /CF/ .$$

Substituting from equation (A21) we find

$$Ut_s = x_o \sec\psi - q/\cot \Theta + \tan\psi/ . \quad \text{(B1)}$$

The maximum value of t_s is given by

$$Ut_s = x_o \sec\psi . \quad \text{(B2)}$$

Category (i) - Interceptions on the feint leg

The maximum allowable take off delay, D, for interception in this category corresponds to interception at the point C in Figure 1.2, so that

$$\frac{/OC/}{V} = \frac{/RC/}{U}.$$

Hence, in terms of t_s, D is given by

$$D = t_s - \frac{/OC/}{V}, \tag{B3}$$

where /OC/ may be expressed as

$$/OC/^2 = (y_T + x_T\tan\psi)^2 + (x_o\sec\psi - Ut_s)^2 - 2(y_T+x_T\tan\psi) . (x_o\sec\psi - Ut_s).\sin\psi . \tag{B4}$$

Category (ii) - Interceptions by fighters reallocated from the feint leg to the second track-leg

In order to calculate where a fighter reallocated from a planned interception on the feint leg achieves, if practicable, an interception on the second track leg, the raid and fighter positions at the track-change point must first be calculated. Assuming a fixed fighter response time to the track-change information, criteria under which the fighter can intercept the raid before it reaches the target may then be determined. For simplicity, we shall use the variable t_s rather than Θ. At the track-change point, from Figure B2 the raid is at the point C with coordinates (x_s,y_s), where

$$x_s = x_o - Ut_s \cos\psi$$

and

$$y_s = y_o + Ut_s \sin\psi . \tag{B5}$$

Consider a fighter scrambled to intercept the raid on its feint leg, with a take-off delay D after initial GC detection of the raid; if we write

$$\tilde{t}_s = t_s - D , \tag{B6}$$

then the fighter is at the point F with coordinates (x_f,y_f) when the raid changes track, where

$$x_f = V\tilde{t}_s \cos\phi$$

and

$$y_f = V\tilde{t}_s \sin\phi .$$

Here ϕ is the initial fighter heading to intercept the feint track.

Suppose that the fighter suffers a fixed reaction delay D_c to the track-change, where D_c may be zero. During this time the fighter is assumed to continue along its original course at its constant speed V; it then instantly adopts a new course to intercept the raid on its second track-leg, if practicable. At time $(t-t_s)$ after the track-change the raid is at the point with coordinates (x_r, y_r), where

$$x_r = x_s + \lambda(t-t_s)(0-x_s) = x_s(1-\lambda(t-t_s)) \ , \qquad (B7)$$

$$y_r = y_s + \lambda(t-t_s)(y_T-y_s) \ ,$$

$$\lambda = \frac{U}{(x_s^2 + (y_T-y_s)^2)^{\frac{1}{2}}} \qquad (B8)$$

and

$$0 \leqslant t-t_s \leqslant \frac{1}{\lambda} \ . \qquad (B9)$$

Hence if $\bar{C}\ (\bar{x}_s, \bar{y}_s)$ is the raid position at the moment the fighter responds to the track-change, then

$$\bar{x}_s = x_s\ (1-\lambda D_c),$$

and $\qquad (B10)$

$$\bar{y}_s = y_s + \lambda D_c(y_T-y_s) \ ,$$

provided that $D_c \leqslant \frac{1}{\lambda}$. The corresponding fighter position is $\bar{F}(\bar{x}_f, \bar{y}_f)$, where

$$\bar{x}_f = V\bar{t}_s \cos\phi \ , \qquad (B11)$$

$$\bar{y}_f = V\bar{t}_s \sin\phi$$

and

$$\bar{t}_s = \tilde{t}_s + D_c = t_s + D_c - D \ . \qquad (B12)$$

If we write

$$t_s^* = t_s + D_c \ , \qquad (B13)$$

then the raid is at the point C^* with coordinates (x_r^*, y_r^*) at time $(t-t_s^*)$ after the fighter has adopted its new interception course, where

$$x_r^* = \bar{x}_s\ (1-\mu\ (t-t_s^*)) \ , \qquad (B14)$$

$$y_r^* = \bar{y}_s + \mu(t-t_s^*)\ (y_T - \bar{y}_s) \ ,$$

$$\mu = \frac{U}{(\bar{x}_s^2 + (y_T - \bar{y}_s)^2)^{\frac{1}{2}}} \ , \qquad (B15)$$

and

$$0 \leqslant t-t_s^* \leqslant \frac{1}{\mu} \ .$$

Assuming that after the delay D_c the fighter immediately adopts its new interception course, if practicable, then the corresponding fighter position is at the point F* with coordinates (x_f^*, y_f^*), where

$$x_f^* = \bar{x}_f + V(t-t_s^*)\cos\phi_1 \quad ,$$
$$y_f^* = \bar{y}_f + V(t-t_s^*)\sin\phi_1 \quad , \tag{B16}$$

and ϕ_1 is the interception heading adopted by the fighter against the second track-leg (see Figure B2).

We now consider the conditions under which the fighter can intercept the raid before it reaches the target, restricting attention to the existence of geometrically feasible interception courses. In a more detailed analysis - involving the fighter model for example - the consequences of imposing fuel limitations on the adoption of interception courses could also be examined.

If Z denotes the distance travelled by the fighter at time $t-t_s^*$ after adopting its new course, then

$$Z^2 = (x_f^* - \bar{x}_f)^2 + (y_f^* - \bar{y}_f)^2 = V^2(t-t_s^*)^2 . \tag{B17}$$

In our simplified model interception occurs when the raid and fighter positions coincide, so that at this point (P' in Figure B2) we may replace x_f^* and y_f^* in (B17) by x_r^* and y_r^* to give

$$\{\bar{x}_s(1-\mu(t-t_s^*)) - \bar{x}_f\}^2 + \{\bar{y}_s + \mu(t-t_s^*)(y_T-\bar{y}_s) - \bar{y}_f\}^2 = V^2(t-t_s^*)^2 \tag{B18}$$

Write

$$X = \bar{x}_s - \bar{x}_f$$

and

$$Y = \bar{y}_s - \bar{y}_f \; ; \tag{B19}$$

rewriting equation (B18) in terms of $(t-t_s^*)$ then gives

$$(t-t_s^*)^2(U^2-V^2) - 2\mu(t-t_s^*)(X\bar{x}_s - Y(y_T-\bar{y}_s)) + (X^2+Y^2) = 0 . \tag{B20}$$

From Figure B2, (X,Y) is the position vector of $\bar{C}$ relative to $\bar{F}$ and $(\bar{x}_s,(\bar{y}_s-y_T))$ is the position vector of $\bar{C}$ relative to T. If we define a, c and ω as shown, then

$$\mu(X\bar{x}_s - Y(y_T-\bar{y}_s)) = Uc.\cos\omega \tag{B21}$$

and

$$X^2 + Y^2 = c^2 . \tag{B22}$$

By substituting from equations (B21) and (B22), equation (B20) can be seen to be of the form of equation (A2), so that from equations (A5) and (A6) we may write

$$t - t_s^* = \frac{c}{2U\cos\omega} \qquad (U=V) \qquad (B23)$$

and

$$t - t_s^* = \frac{c}{U^2 - V^2}\{U\cos\omega - (V^2 - U^2\sin^2\omega)^{1/2}\} \qquad (U \neq V) \qquad (B24)$$

where the condition $t - t_s^* < \frac{1}{\mu}$ corresponds to

$$\frac{/\bar{F}T/}{V} < \frac{/\bar{C}T/}{U} . \qquad (B25)$$

In terms of the coordinate system, equation (B20) has solutions

$$t - t_s^* = \frac{X^2 + Y^2}{2\mu(X\bar{x}_s - Y(y_T - \bar{y}_s))} \qquad (U=V) \qquad (B23)'$$

and, if $K \geq 0$,

$$t - t_s^* = \frac{\mu(X\bar{x}_s - Y(y_T - \bar{y}_s)) - \sqrt{K}}{U^2 - V^2} \qquad (U \neq V) \qquad (B24)'$$

where

$$K = V^2(X^2+Y^2) - \mu^2(Y\bar{x}_s + X(y_T - \bar{y}_s))^2 . \qquad (B26)$$

In the limiting case in which interception occurs over the target, we have

$$\frac{/\bar{F}T/}{V} = \frac{/\bar{C}T/}{U}$$

From Figure B2, this gives

$$(V\bar{t}_s - y_T\sin\phi)^2 = \frac{V^2}{U^2}a^2 - y_T^2\cos^2\phi ,$$

so that

$$V\bar{t}_s = y_T\sin\phi \pm \left(\frac{V^2}{U^2}a^2 - y_T^2\cos^2\phi\right)^{\frac{1}{2}} , \qquad (B27)$$

where from (B12)

$$\bar{t}_s = t_s + D_c - D$$

and

$$a^2 = /\bar{C}T/^2 = \bar{x}_s^{\,2} + (y_T - \bar{y}_s)^2 .$$

Application

Although ϕ has been referred to as the heading of the initial fighter interception course, this assumption is not used in the analysis and indeed ϕ may be an arbitrary initial fighter heading. We consider here a scramble policy which enables fighters to intercept successfully, although the initial fighter heading ϕ need not correspond to an interception course against the raid on its feint leg. We concentrate on the limiting case of interceptions directly over the target, and conditions are found whereby the point at which the raid abandons its feint track does not affect the fighter intercept capability. For simplicity, we suppose that the fighter reaction delay to the track-change is zero, so that

$$D_c = 0 \ . \qquad \text{(B28)}$$

Referring to Figure B2, suppose we choose ϕ in equation (B27) such that

$$\frac{|y_T \cos\phi|}{V} = \frac{q}{U} \ . \qquad \text{(B29)}$$

This gives

$$V\tilde{t}_s = y_T \sin\phi \pm \frac{V}{U} \cdot (a^2-q^2)^{\frac{1}{2}} \ , \qquad \text{(B30)}$$

where

$$\tilde{t}_s = \bar{t}_s = t_s - D \ .$$

Now

$$|RC| = U\tilde{t}_s = \frac{U}{V}\,(V\tilde{t}_s) = \frac{U}{V}\,(y_T \sin\phi \pm \frac{V}{U}\,(a^2-q^2)^{\frac{1}{2}}) . \qquad \text{(B31)}$$

Also

$$\begin{aligned} |RC| &= |RN| \pm |NC| \\ &= |RN| \pm (a^2 - q^2)^{\frac{1}{2}} \ , \end{aligned} \qquad \text{(B32)}$$

where the positive sign is taken if the track-change occurs before the point N ($\Theta < \frac{\pi}{2}$) and the negative sign if it occurs after the point N ($\Theta > \frac{\pi}{2}$).

From equations (B31) and (B32) we have

$$\begin{aligned} \frac{U}{V}\, y_T \sin\phi &= |RN| \\ &= (x_o \sec\psi - x_T \sin\psi \tan\psi) - UD \ , \end{aligned} \qquad \text{(B33)}$$

so that

$$D = \frac{1}{U}\,(x_o \sec\psi - x_T \sin\psi \tan\psi) - \frac{U}{V}\, y_T \sin\phi \ . \qquad \text{(B34)}$$

Equation (B34) shows that we have eliminated the point at which the raid changes track from the expression for fighter intercept capability, as measured by the maximum take-off delay D for scramble and eventual interception over the target.

Figure B3 illustrates this application for the case U = 2V, with the standard raid feint pattern, case (1). From the preceding analysis, without loss of generality we may take a raid track-change angle of $\Theta = \frac{\pi}{2}$. For a given feint angle ψ, a circle of radius r is drawn centred on the target T, where

$$r = q.\frac{V}{U} . \qquad \text{(B35)}$$

The tangent to this circle through the origin (i.e., the fighter base) is then drawn as shown. This defines the initial fighter heading ϕ against the feint track, with

$$/y_T\cos\phi/ \;=\; r \;=\; q.\frac{V}{U} , \qquad \text{(B36)}$$

as specified in equation (B29).

Note that the fighter track given by the heading ϕ does <u>not</u> correspond to an interception course against the feint track, for a fighter which takes off when the raid is at the point R. Indeed, since the angle α is such that

$$\alpha > \sin^{-1}(\frac{V}{U}) = 30^o ,$$

a fighter taking off when the raid is at the point R cannot intercept the raid on its feint track. The point R' shown in Figure B3 denotes the raid position at fighter take-off if the fighter heading ϕ is to correspond to an interception course against the feint track. Hence although a fighter with a delay before take-off such that the raid is at the point R when it scrambles cannot intercept on the feint track, it can still intercept the raid over the target, regardless of the point at which it abandons its feint track.

Such a scramble policy does not of course allow as great a fighter take-off delay (or as many fighters scrambled against the raid) as scramble policy III; under this policy the maximum take-off delay for interception is such that the raid is at the point R" (see Figure B3) when the fighter scrambles. As with policy III, this policy implies some knowledge of the raid target T, since its distance y_T along the target axis from the fighter base is used in the calculation of ϕ (equation (B29)). Nevertheless, it serves to emphasise the value to the fighter defences of a more 'liberal' scramble policy, under which fighters are prepared to scramble without close regard to the estimated position of interception.

It should be noted that, at the level at which this analysis is set, the possible disadvantages of such scramble policies are not considered here. These include, for example, the situation in which fighters are out of action at base for refuelling after an 'optimistic' scramble when further raids penetrate their specified area of cover.

Finally, Figure B4 presents a similar illustration of interception over the target, in the case of equal fighter and raid speeds, i.e., $U = V$; this special case is examined in more detail in Chapter 2 of the main text. Equation (B27) is illustrated by showing the interceptions over the target corresponding to arbitrary track-change angles Θ and $(\pi-\Theta)$.

In this case the maximum fighter heading ϕ at take-off for successful interception derived from (B36) not only guarantees an interception over the target regardless of where the track-change occurs, but easily it also corresponds to an interception course against the raid feint track. Thus, in the notation of Figures B3 and B4,

$$R = R' \ .$$

Hence the maximum acceptable take-off delay such that a fighter scrambled onto an interception course against the feint track can still intercept the raid before or over the target is such that the raid is at the point R when the fighter takes off; the corresponding fighter heading is denoted by ϕ, given by (B36). Further, this result holds independently of where the raid track-change occurs.

Category (iii) - Interceptions by fighters scrambled against the second track-leg

Interceptions under scramble policy III may be regarded as in this category, with the maximum take-off delay for successful interception given by equations (A22) and (A24); the relationship between take-off delay D, relative to first detection, and the serial I of the corresponding fighter scrambled is given by equation (1.5). Under the other scramble policies it is assumed that, if the information processing delay D_p has elapsed between first detection of the raid at the point S and its track-change, then there is no further such delay imposed on fighters scrambled after the track-change, merely the delay D_r representing their readiness level. If D' denotes the maximum allowable take-off delay, relative to the time of the track-change, and I is the serial of the corresponding fighter scrambled (and, easily, the corresponding number of interceptions achieved), then D' and I are related by

$$D' = D_r + I.D_s \ . \quad (I = 1,2,3,\ldots\ldots) \tag{B37}$$

The maximum take-off delay for successful interception is most easily expressed in terms of Θ, and from Figure 1.2 this is given by

$$\frac{U}{V}\, y_T = q/\mathrm{cosec}\Theta/ - UD' \ ,$$

i.e.

$$D' = \frac{1}{U}\left(/y_M \cos\psi \,\mathrm{cosec}\Theta/ - \frac{U}{V}\, y_T\right) . \tag{B38}$$

If the raid changes track before the information processing delay D_p is complete, or if it is not detected until after it changes track, then the maximum allowable take-off delay is again given by equations (A22) and (A24) respectively.

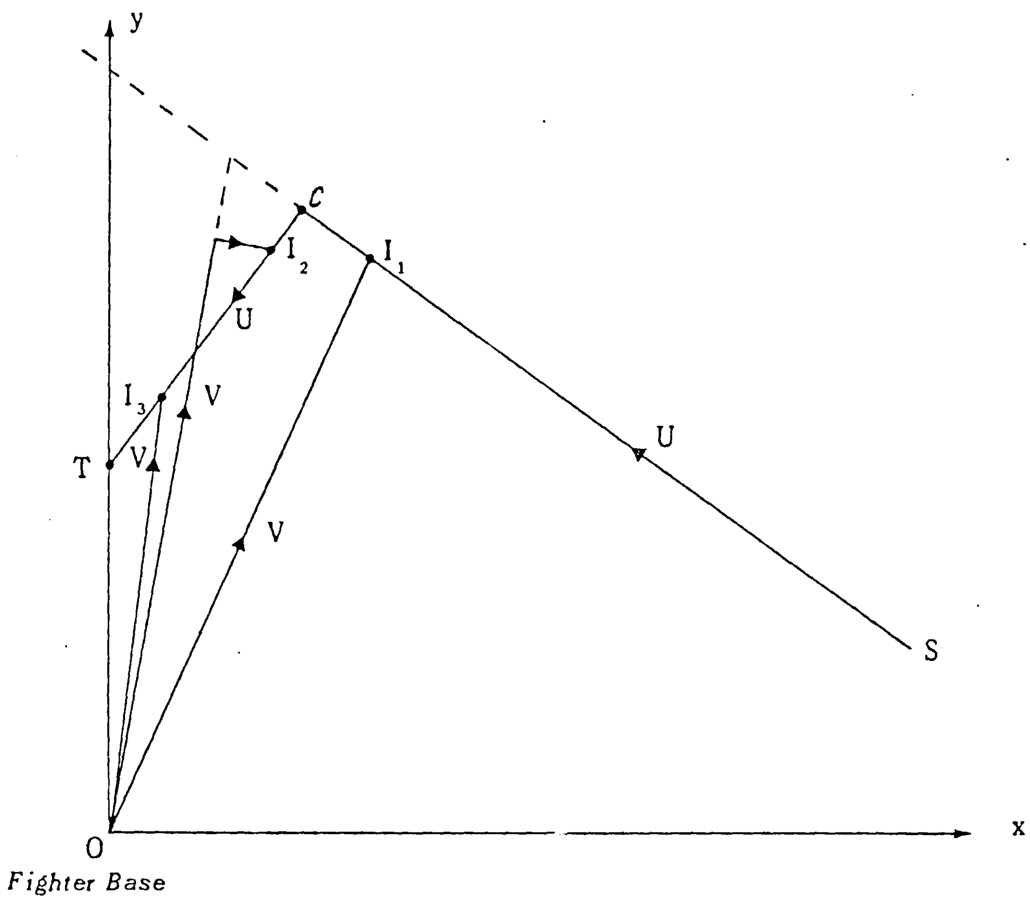


FIGURE B1 THREE TYPES OF INTERCEPTION

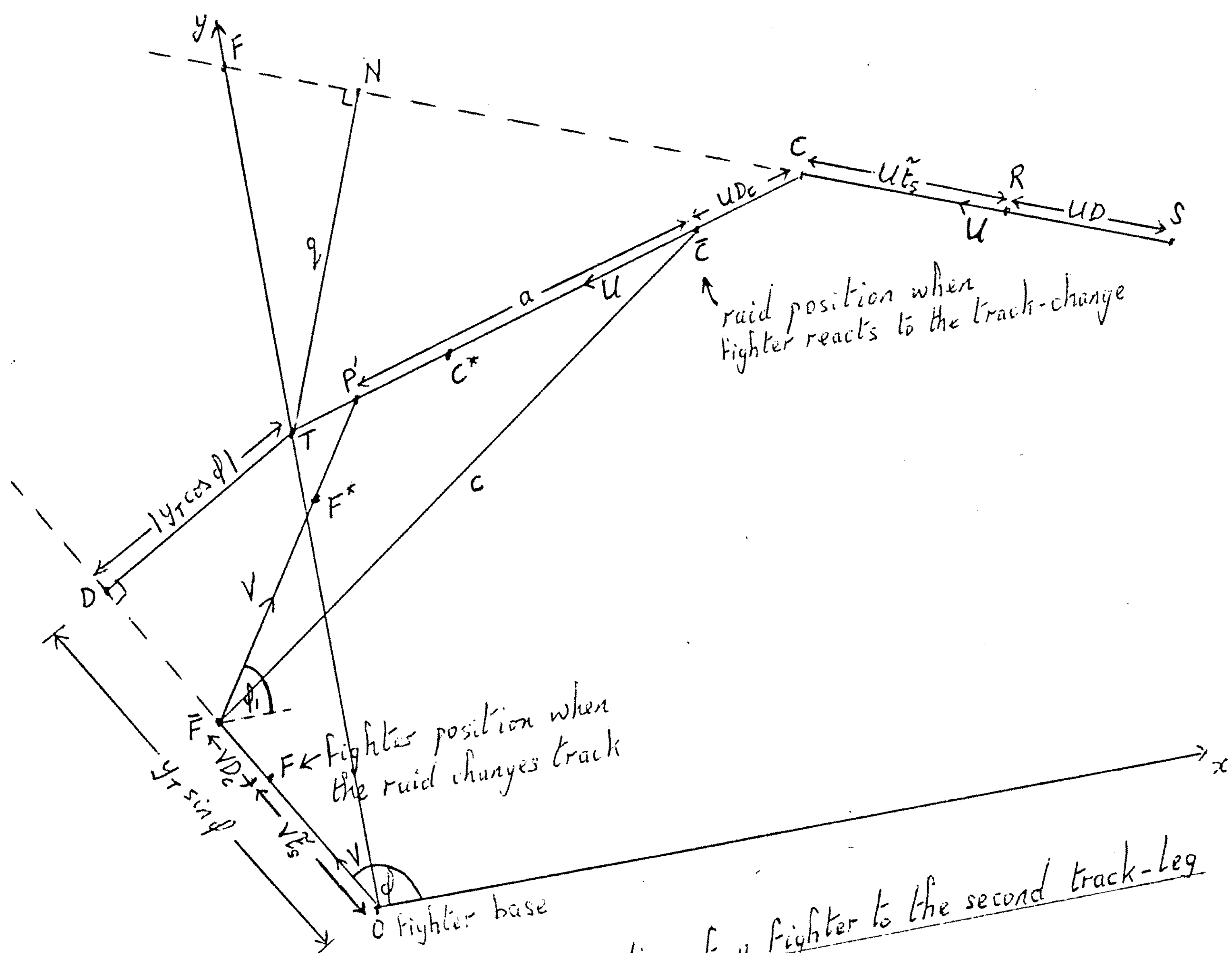


Figure 52 Reallocation of a fighter to the second track-leg

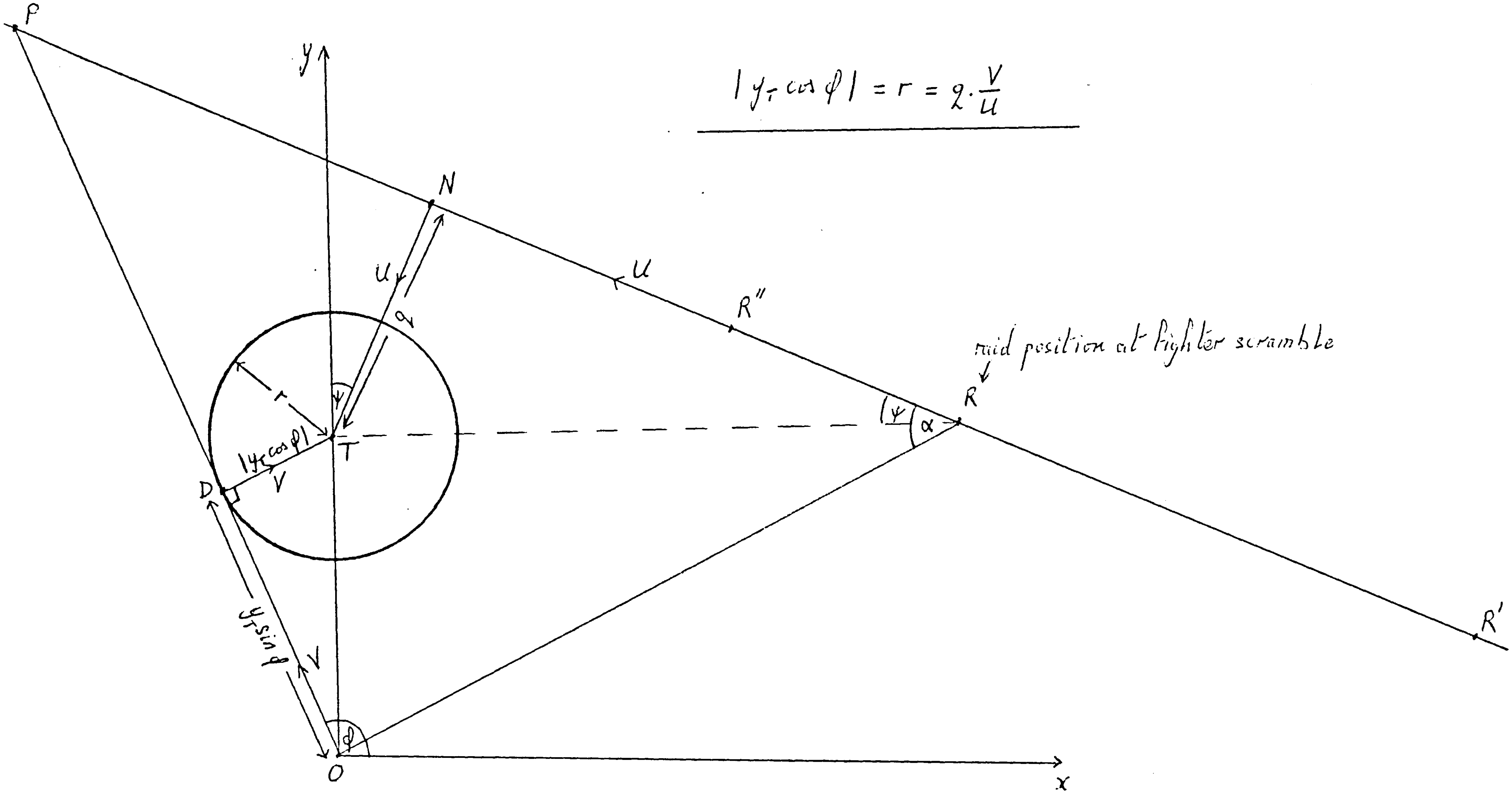
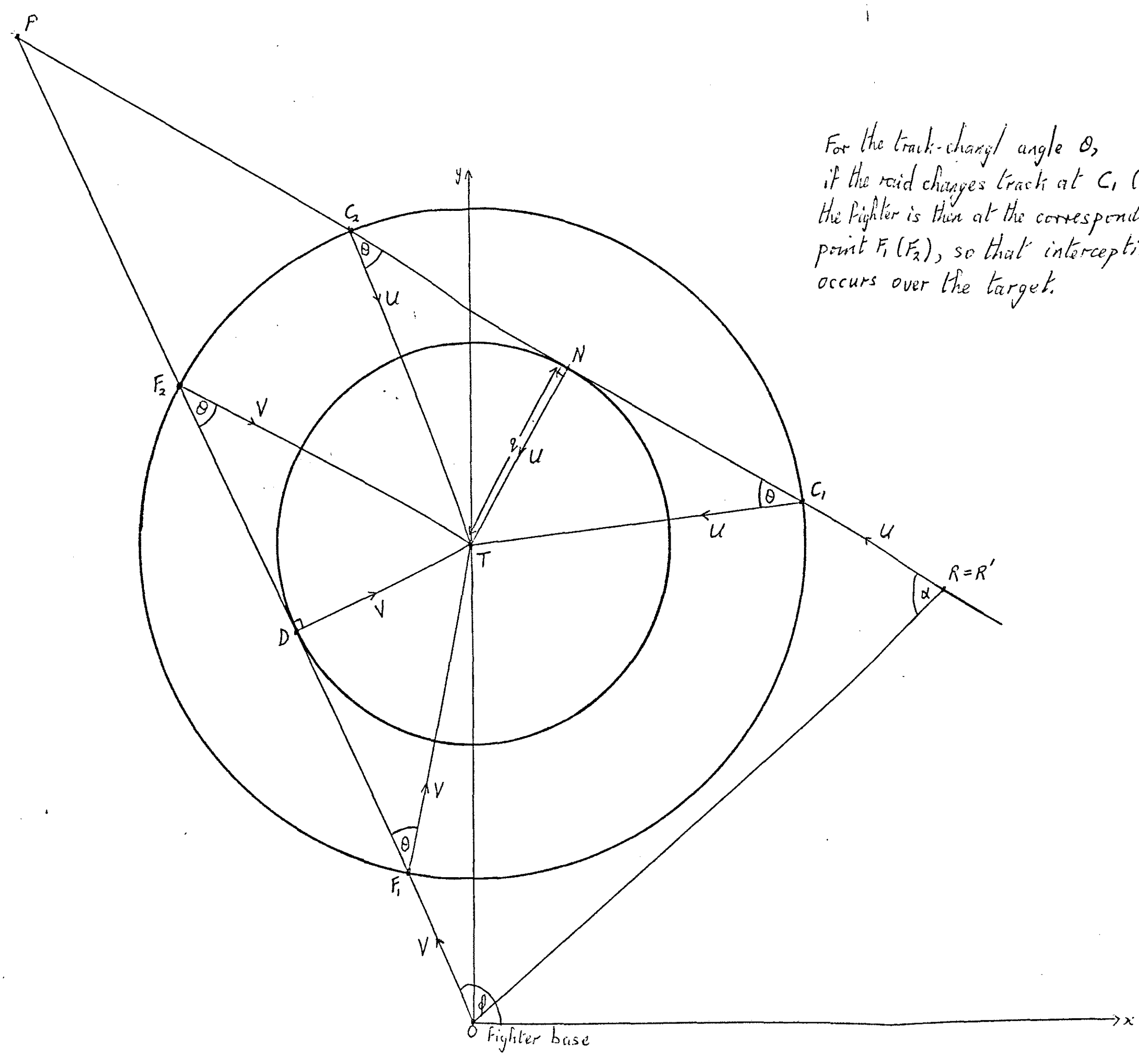


Figure B3 Interception over the Target : U=2V



For the track-change angle θ, if the raid changes track at C_1 (C_2), the fighter is then at the corresponding point F_1 (F_2), so that interception occurs over the target.

Figure B4 Interception over the Target: U = V

PART 4

THE EFFECTS OF
SENSOR INFORMATION ERRORS ON
FIGHTER INTERCEPT CAPABILITY

CONTENTS OF PART 4 - THE EFFECTS OF SENSOR INFORMATION ERRORS ON FIGHTER INTERCEPT CAPABILITY

APPENDIX A OF PART 4 - MATHEMATICAL ANALYSIS OF SENSOR INFORMATION ERRORS

1 INTRODUCTION

In this Part of the thesis we present a fairly detailed mathematical analysis of the effect of sensor errors on fighter intercept capability. Errors in the initial Ground Control (GC) estimates of raid position, speed and heading at initial detection are considered. A pessimistic case for the fighter defences is examined, in which the initial and possibly erroneous GC estimate of the raid track is not updated, with presumably more accurate information, as the raid continues its approach. This is not a very realistic assumption, but it does provide a 'worst case' background for the analysis. As mentioned later, the fighter model (see Part 2) was also used a great deal in this study. An assessment of the effect of regular (or irregular) updating of GC information on fighter intercept capability would be better suited to the fighter model simulation. To extend this analysis for the purpose would push it beyond the bounds of acceptable complexity.

The studies of sensor information errors and raid indirect routing are both derived from DOAE Study 250. The relevance of these two aspects of fighter intercept capability is assessed separately, so that in the analysis of GC sensor information errors we assume that raids do not engage in indirect routing but fly directly towards their targets. Otherwise the input assumptions underlying these two studies are the same - a single fighter base defending a single target against attack from an overflying, concentrated (point) raid; low-level attacks at high subsonic speeds, etc. - and we refer to Chapter 1 of Part 3 for details. In particular, Figure 1.1 of Part 3 illustrates the derivation of the coordinate system which is used in the numerical examples presented later. This enables us to introduce a concept of 'area defence' by fighters, rather than 'point defence' of particular, well-defined targets. Thus we require that, for a fighter interception to be deemed successful, it must occur before the raid crosses the y-axis, which is also termed the target axis. For the purpose of obtaining a broad assessment of fighter effectiveness, we may then regard the raid target as being at the intersection of its track with the target axis, and interceptions must occur before the raid reaches this point.

2. SCOPE

The analysis is first developed in Annex A, in which the measure of fighter effectiveness is the maximum acceptable systematic error in the estimated raid track for interception by a single fighter, expressed as a function of the timeliness of the warning provided by GC. Systematic errors in the GC sensors could correspond, for example, to biases in the detection and information extraction subsystems. The analysis is then extended, by use of the fighter model, to give values of mean fighter intercept effectiveness for random errors with a known distribution (see Appendix B). To illustrate this, we present the fighter effectiveness as a function of the standard deviation of normally distributed errors with zero mean.

The GC estimates of raid position, speed and track at first detection are not updated, and the fighters must themselves detect the raid before they correct their courses (a pessimistic condition). Errors in estimating the initial raid coordinates, speed and track are studied independently. The numerical examples apply only to raid tracks which are direct and perpendicular to the target axis. Finally, no attempt is made to study the long-term consequences of errors by GC in estimating the raid size, i.e., the number of aircraft in the raid.

There is no explicit representation of jamming in this study, whether by stand-off jammers, self-screening or escort jammers. Nevertheless the magnitude of the errors in GC sensor information considered in the numerical examples presented is sufficiently great so as to correspond to fairly severe jamming of GC surveillance radars in support of the raid. It is also assumed that the fighters can respond quickly when they detect the raid and that they maintain their detection capability during attempted interceptions. However, if their AI radars are jammed, in either search or tracking mode, these assumptions will generally not hold, unless the fighters can carry out some form of rapid and accurate bearings - only analysis. For example, this could be by triangulation of the jamming strobes, using data-links between fighters. Conversely, if fighters must resort, in the face of AI radar jamming, to some form of kinematic ranging, in which large changes in course are required in order to measure the rate of change of bearing of the raid, then their ability to react quickly and effectively when they detect the raid may be seriously degraded.

3. ANALYTICAL METHODS

Both the mathematical analysis described in the Appendices and the fighter model described in Part 2 are used in the study of information errors. The mathematical analysis is used to derive expressions for the maximum acceptable errors which still enable a fighter to scramble, detect and intercept the raid before it reaches the target axis, for a fixed fighter take-off delay (corresponding, in the numerical examples presented, to the first fighter scrambled). Given the number of fighters assumed available at the base, the fighter model is then used to derive the effect of information errors in terms of the standard measure of effectiveness, namely the total number of interceptions achieved before the raid reaches the target axis. Of the scramble policies considered in the study of raid indirect routing, only policies I and II have practical application here. Scramble policy III need not be considered, for if the point at which the raid is expected to cross the target axis is known, GC sensor errors may be ignored. In addition, scramble policy IV (a fuel-limitation policy) has little effect in the numerical examples considered, although for completeness the maximum allowable GC sensor errors for scramble under this policy are derived in Appendix A. The dominant factors determining fighter intercept capability are the amount of warning available to Ground Control, the fighter detection capability and the time taken by fighters to react to a detection. The fighter endurance capability when scrambled from ground alert with a realistic fuel load is rarely a limiting factor in the scenarios examined in this study.

We assume fighters have a deterministic radar detection capability, defined by the radar range and angle of scan. Within the sector of the circle defined by these parameters a fighter detects with 100% probability, while it has zero probability of detection outside this sector. We consider the problem of fighter detection from a geometrical standpoint only; the fighter model could be used to investigate the effect of pulse and pulse doppler clutter on the fighter detection capability.

An extra factor is incorporated in the fighter model study which is difficult to represent analytically. In the fighter model, it is assumed that if a fighter reaches its expected interception point without achieving a detection, it turns outbound (parallel to the x-axis and in the direction of x increasing), towards a holding Combat Air Patrol position (CAP), where it continues to seach for the raid. In the mathematical analysis the fighter is assumed to continue indefinitely along its initial course. This track-change is generally advantageous to the fighters, improving their chances of detecting and subsequently intercepting the raid.

4. NUMERICAL INPUTS

The numerical inputs used in the specific examples presented are summarised in Table 1. As in the study of raid indirect routing,the point at which the raid crosses the target axis, i.e., the point which under our area defence system we regard as the raid target, is half-way between the fighter base considered and its nearest neighbour along the target axis.

5. RESULTS

The results are presented in Figures 1-8, where the GC estimates of the raid's x- and y- coordinates at first detection are termed the estimated warning distance and offset respectively. Figures 1-4 are derived analytically, and show the maximum acceptable error in each of the raid parameters, for interception by the first fighter scrambled. The maximum error is shown as a function of the actual warning distance achieved against the raid, i.e. the perpendicular distance of the raid from the target axis when it is first detected. The limitations of adopting scramble policy I rather than policy II are also shown. The derivation of the various components of these figures is given in Appendix A.

The fighter model was used to determine the number of interceptions achieved as a function of the error in the initial GC estimate of the raid parameters, for a fixed warning distance of 360 nm; these results are presented in Appendix B. Figures 5-8 present the corresponding results for random (normal) errors, assumed normally distributed with zero mean. The number of interceptions achieved is shown as a function of the standard deviation of the error in the estimate of each of the raid parameters. A maximum of 20 fighters available is assumed, operating under scramble and control procedure II, modified as explained in Chapter 3.

6. DISCUSSION

The results show the relatively large errors which can be tolerated in the initial estimates of the raid parameters, when an AI radar detection range of about 90 nm is available and the fighters can respond quickly when they detect the raid. For example, Figure 1 shows the acceptable error in the estimated warning distance of the raid at first detection. Although no error can be tolerated if the raid is not detected until it is 200 nm from the target axis, if this warning distance is increased to, say, 240 nm, this can be estimated at as much as 500 nm or as little as 200 nm before the first fighter scrambled fails to intercept. If scramble policy I is adopted rather than policy II, the acceptable underestimate in the raid's warning distance is only reduced for warning distances less than 230 nm. Thus, as in the indirect routing study, with sufficiently large warning distances the significance of the scramble policy is diminished. Figure 1 also shows that an overestimate of the distance of the raid at first detection is less serious than a corresponding underestimate.

Figure 2 shows the corresponding results for the estimated raid offset at first detection. Again, realistic errors in the estimated raid offset can be tolerated for reasonable warning distances. The actual raid offset is 108 nm; with a warning distance of only 240 nm this can be estimated at between 144 nm and 45 nm before the first fighter scrambled fails to intercept. With this warning distance, the acceptable error in estimating the raid offset is unaltered if scramble policy I is adopted.

Figure 3 shows the acceptable error in the estimate of the raid heading, defined to be the angle which the raid track is believed to make with the x-axis. The actual direction of the raid is defined as 0^{o}. Again, realistic errors in estimating this parameter do not affect the interception capability of the first fighter scrambled. With the above warning distance the raid heading can be estimated to be $+10^{o}$ or -20^{o} before interception fails. With the same warning distance of 240 nm, Figure 4 shows that the raid speed of 9 nm/min can be overestimated by 1.5 nm/min (90 knots), or enormously underestimated before the first fighter scrambled fails to intercept.

Figures 5-8 apply to the case when very good early warning is available to the fighter defences. Together with an AI detection range of 90 nm, this prevents any realistic random error in the estimates of any of the raid parameters from seriously degrading the fighter intercept capability. For example, a standard deviation of 200 knots in estimating the raid speed, or 100 nm in estimating the raid's warning distance, decreases the mean number of possible interceptions by 1. A standard deviation of 10^{o} in estimating the raid heading decreases the mean number of interceptions by 1, as does a standard deviation of 30 nm in estimating the raid offset.

7. CONCLUSIONS

If a reasonable warning distance can be achieved realistic errors in estimating the raid parameters do not degrade the fighter intercept capability, provided the range at which the fighters detect is not itself seriously degraded. The warning distance required depends on the number of fighters required to intercept the raid. To achieve 20 interceptions from a single fighter base against a raid on a perpendicular track to a target offset by about 100nm, errors in estimated offset distance and raid heading appear to be potentially the most serious. The assumed ability of the fighter radar to maintain its detection capability (by bearings-only analysis against self-screening jamming), together with the fighters' assumed ability to respond quickly to a detection, contribute significantly to their ability to tolerate GC sensor information errors.

TABLE 1
NUMERICAL INPUTS TO THE STUDY OF SENSOR INFORMATION ERRORS

	RAID PARAMETERS	
(A)	Target	108nm along the target axis
(B)	Track	Direct track to the target, perpendicular to the target axis
(C)	Raid speed (constant)	9nm/min
(D)	Raid length - concentrated raid	zero
	GC AND FIGHTER PARAMETERS	
(A)	Actual warning distance at first detection of the raid	arbitrary
(B)	Sensor & GC processing and transmission delay at first detection	5 minutes
(C)	Fighter readiness level	5 minutes
(D)	Scramble Policy: I : Scramble if expect to intercept before the target axis II : Scramble unless interception of estimated track appears impossible	
(E)	Delay between take-offs	30 secs (ie scramble rate = two per minute)
(F)	Number of fighters available at the base	twenty
(G)	Fighter speed (constant)	9nm/min
(H)	Delay between a fighter detecting the raid and correcting its interception course	zero
(I)	Initial GC estimate of raid speed Initial GC estimate of raid heading Inital GC estimate of raid position: (i) x-coordinate (warning distance) (ii) y-coordinate (offset)	arbitrary " arbitrary "
(J)	Fighter radar range	90nm
(K)	Fighter radar angle of scan	$\pm 60^{o}$

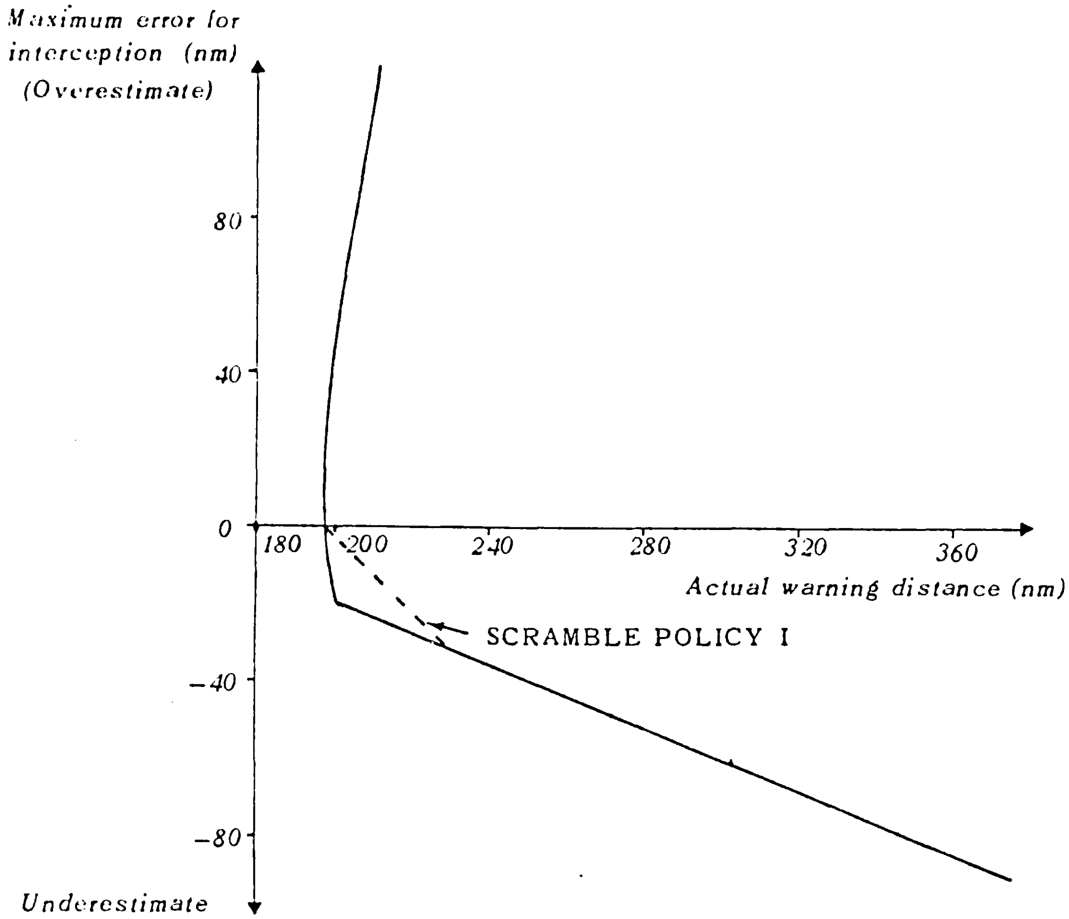


FIGURE 1 MAXIMUM ERROR IN THE ESTIMATED WARNING DISTANCE AT FIRST DETECTION

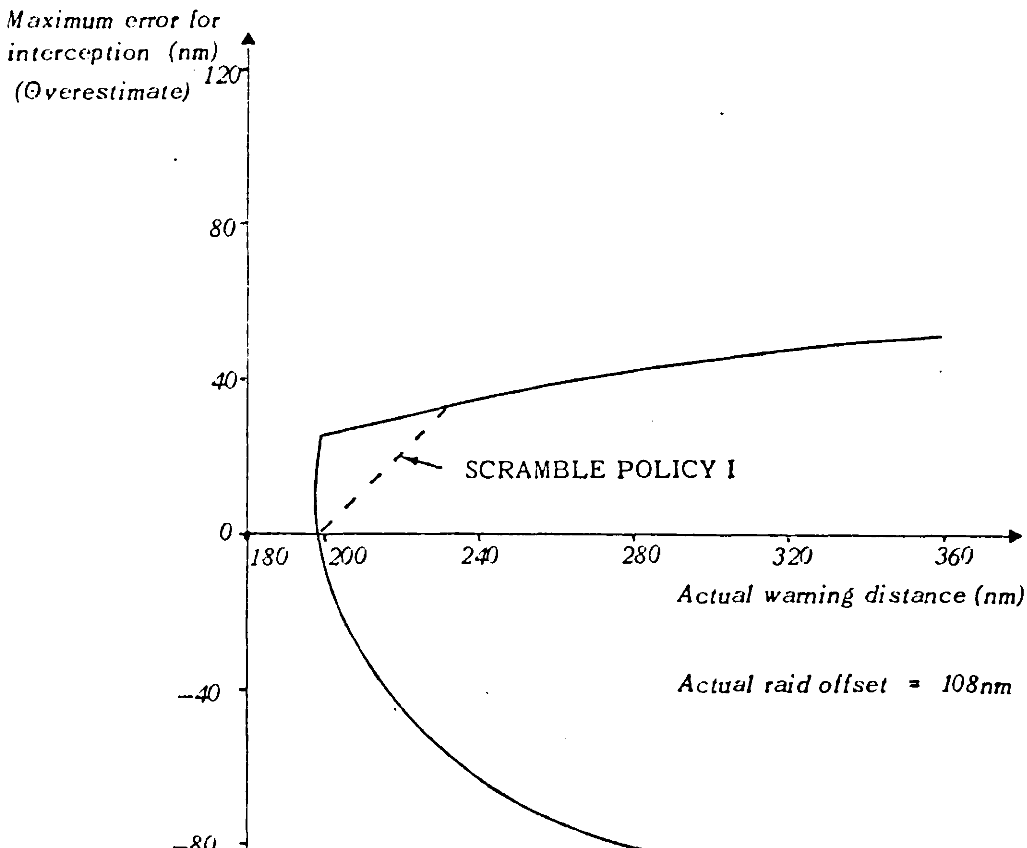


FIGURE 2 MAXIMUM ERROR IN THE ESTIMATED RAID OFFSET AT FIRST DETECTION

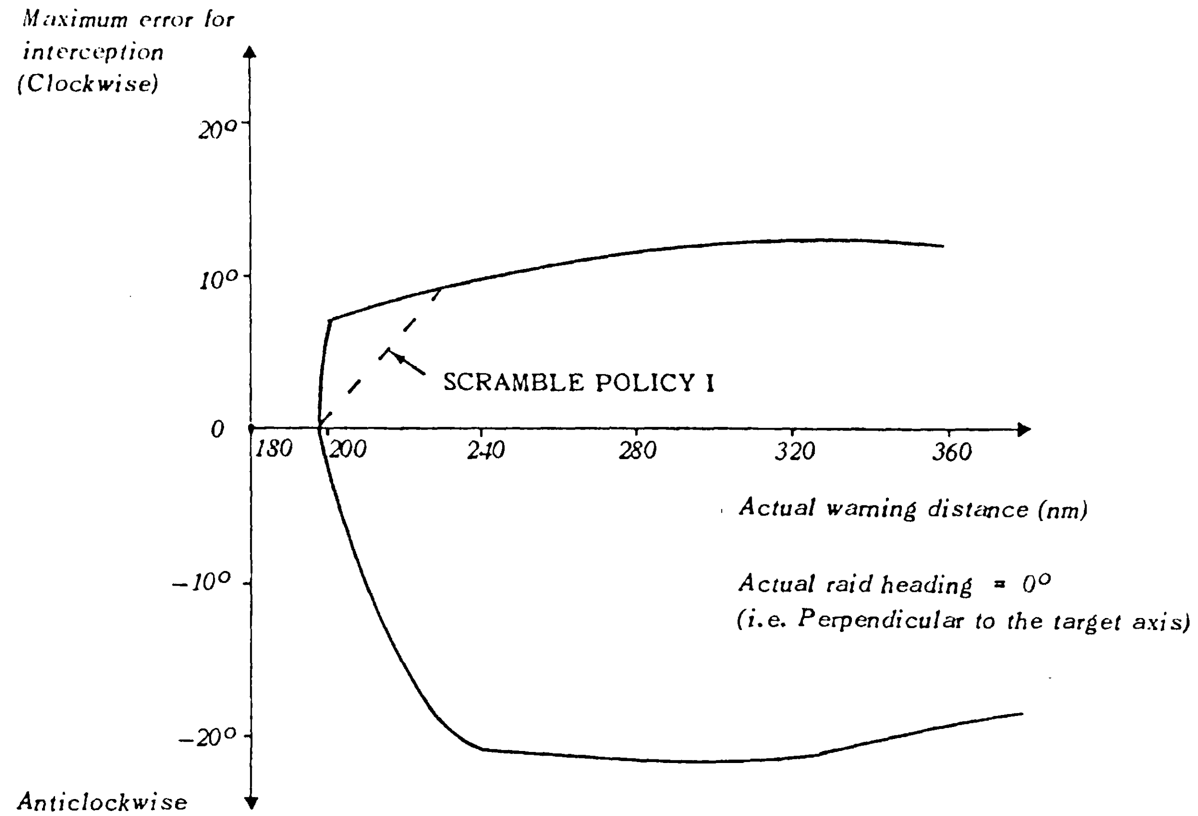


FIGURE 3 MAXIMUM ERROR IN THE ESTIMATED RAID HEADING AT FIRST DETECTION

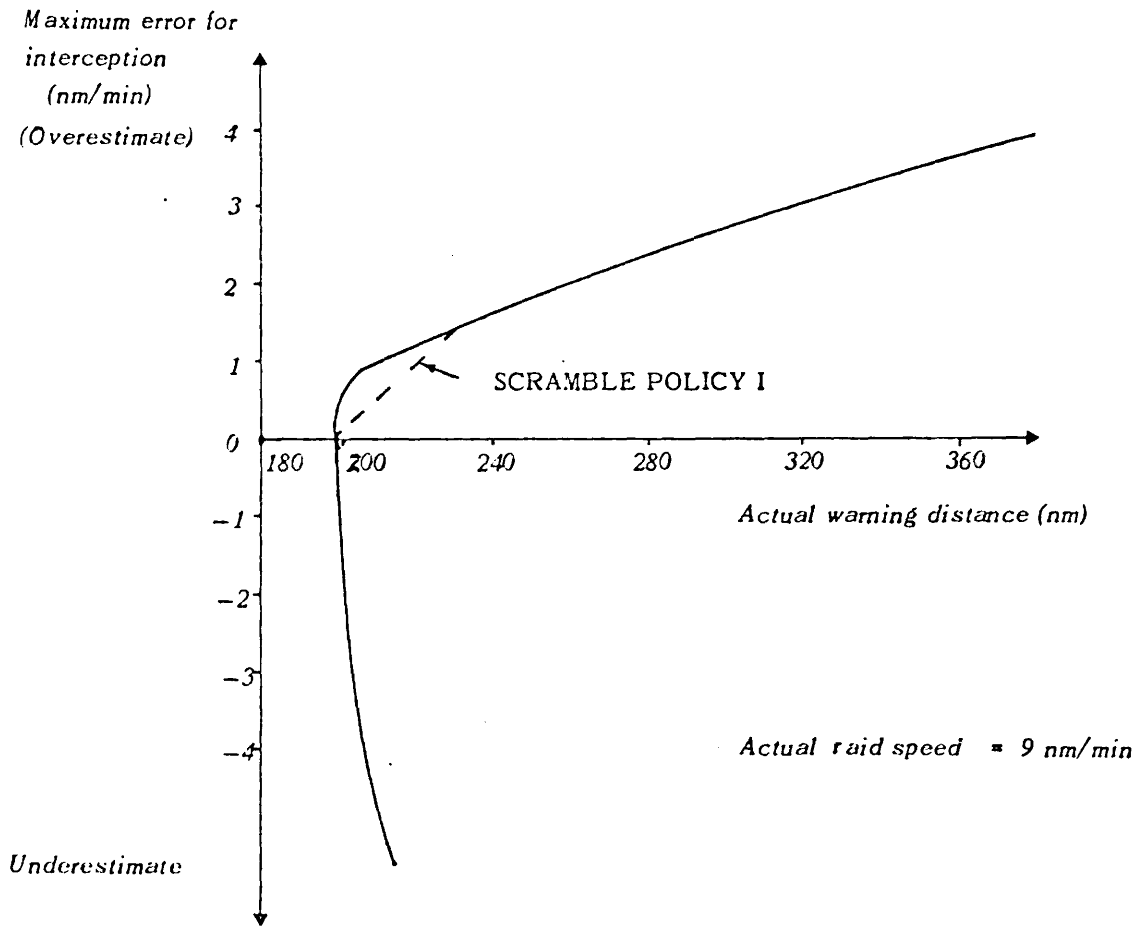


FIGURE 4 MAXIMUM ERROR IN THE ESTIMATED RAID SPEED AT FIRST DETECTION

Normally distributed errors in the G.C. estimate of the raid parameters at first detection

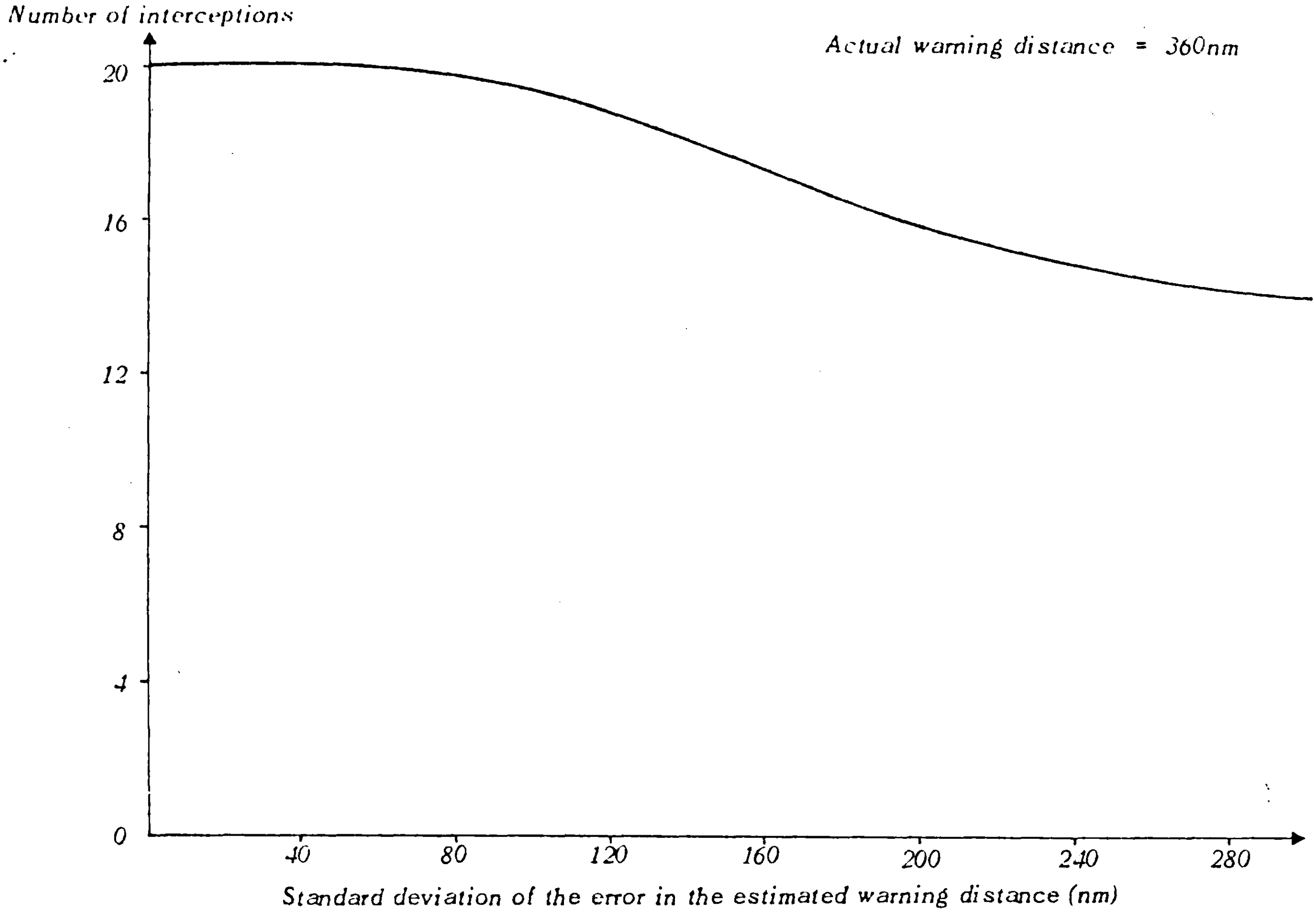


FIGURE 5 WARNING DISTANCE

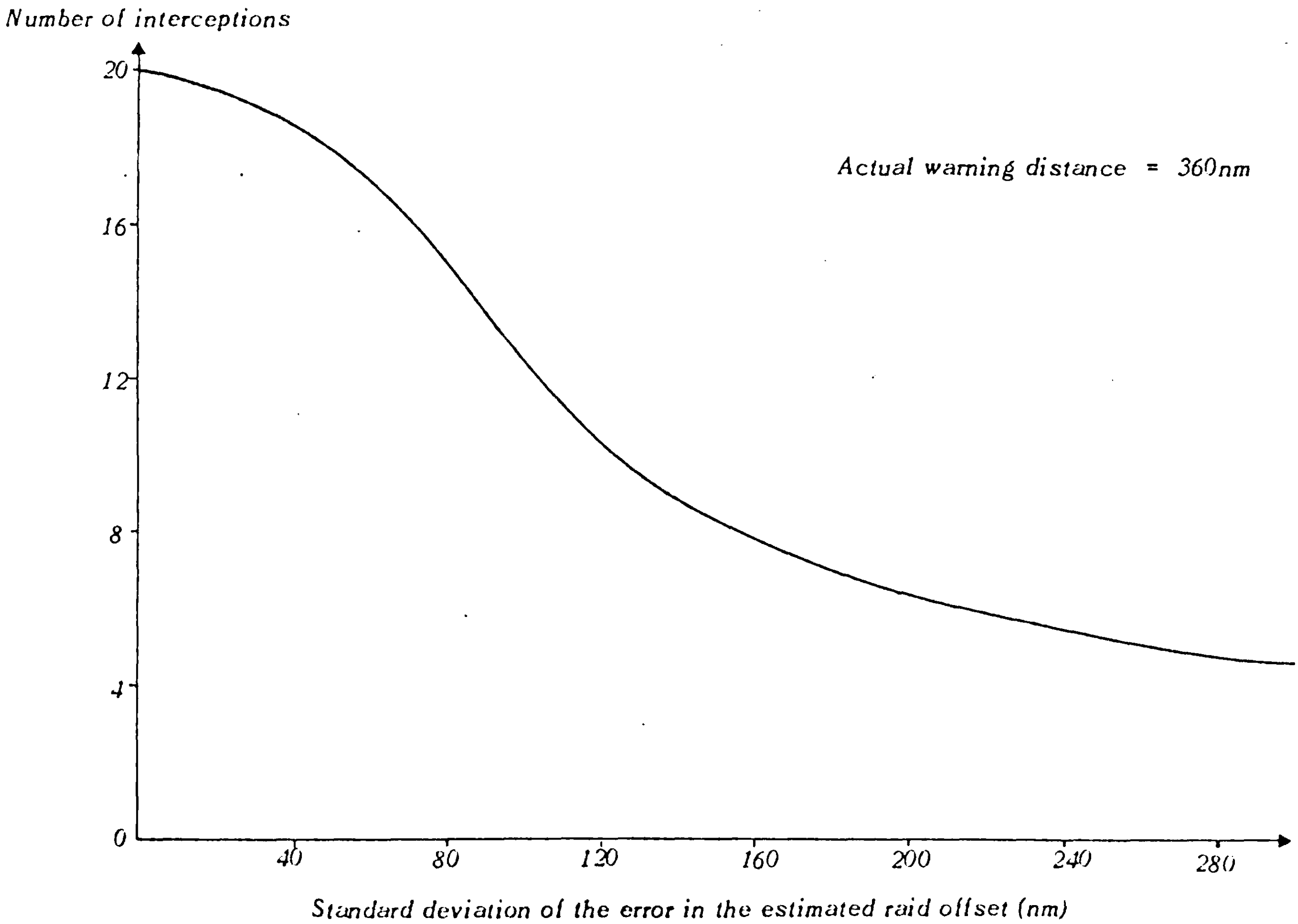


FIGURE 6 RAID OFFSET

Normally distributed errors in the G.C. estimate of the raid parameters at first detection

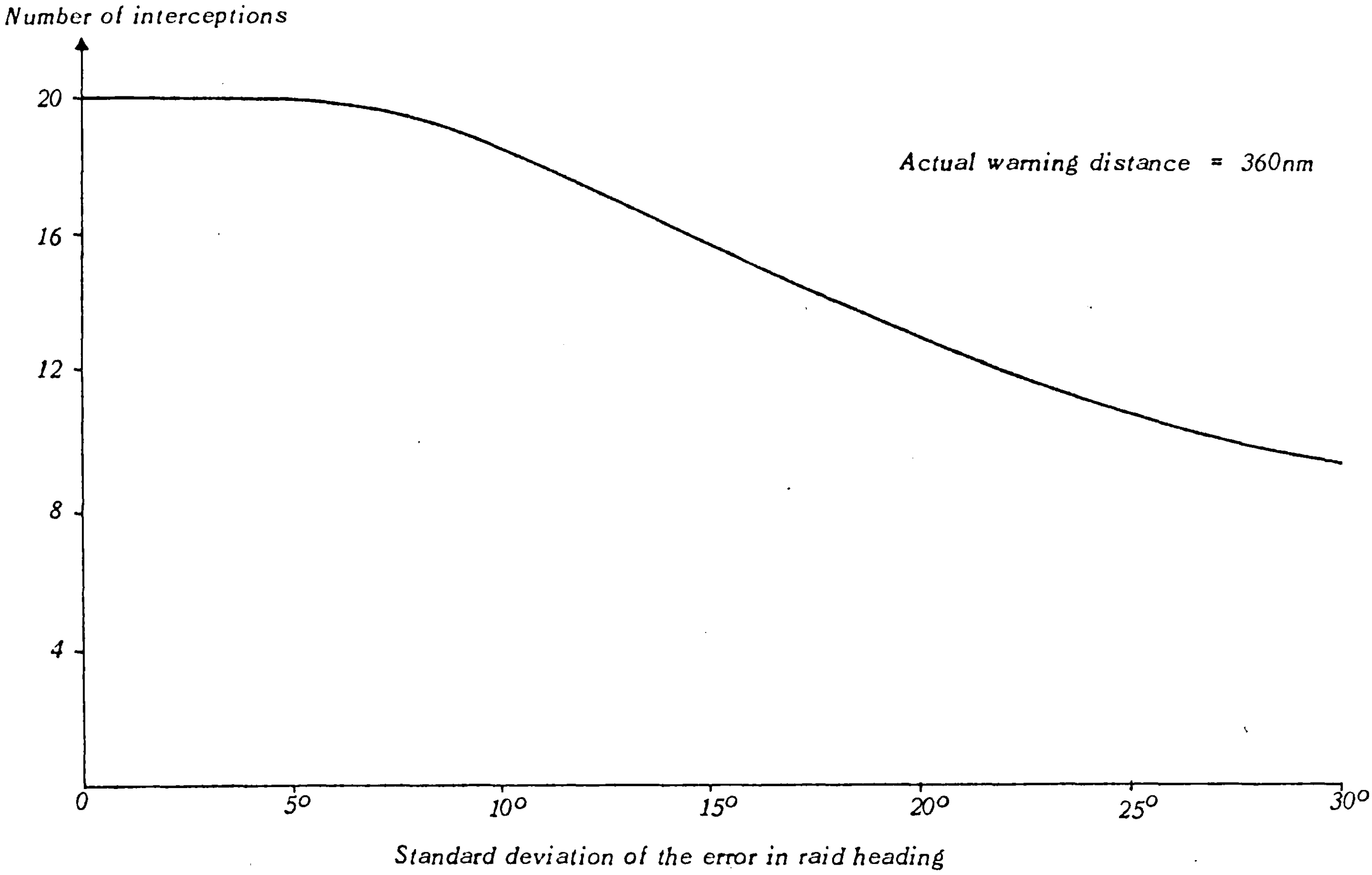


FIGURE 7 RAID HEADING

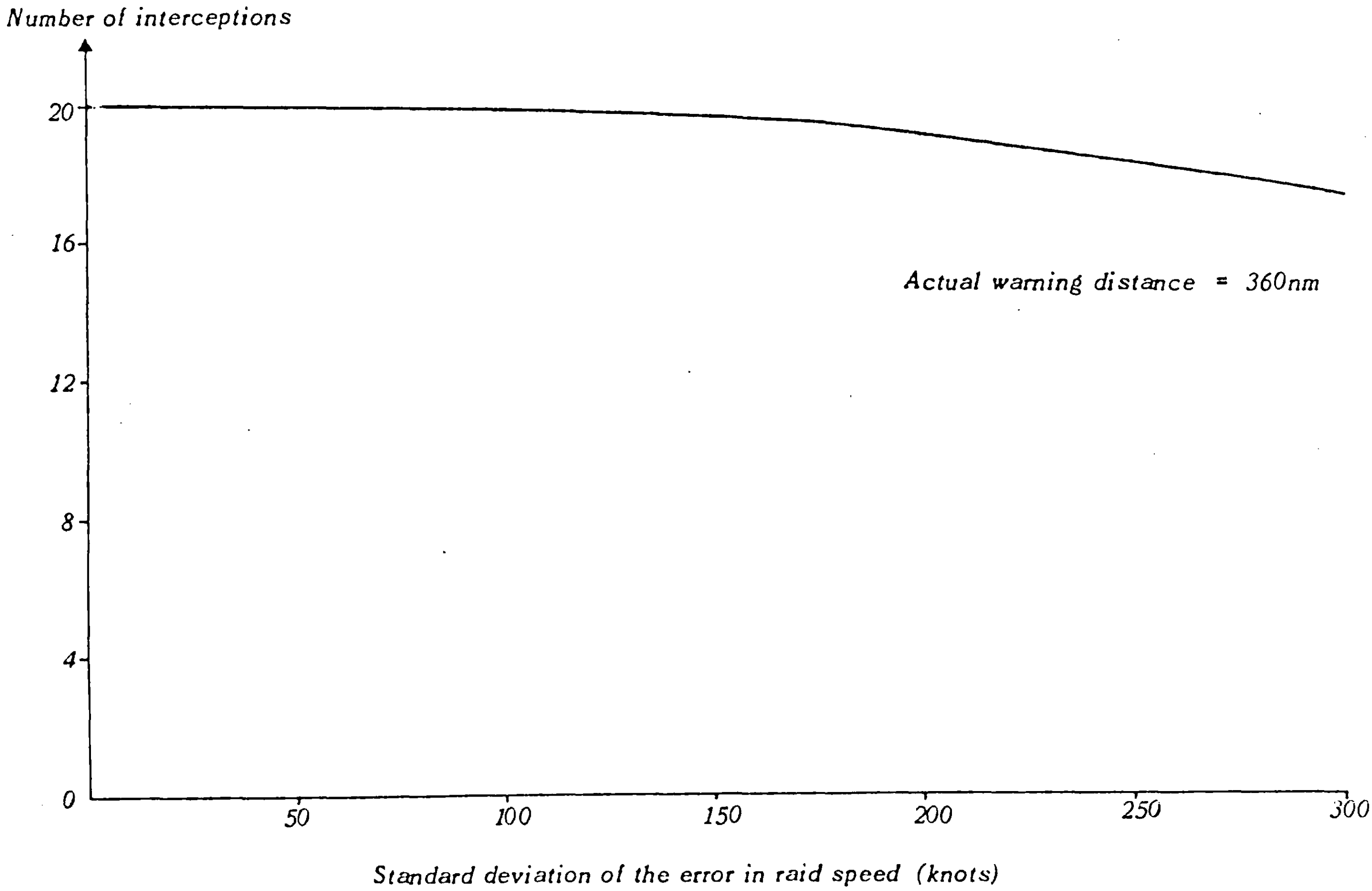


FIGURE 8 RAID SPEED

APPENDIX A

MATHEMATICAL ANALYSIS

1. INTRODUCTION

This Appendix studies the effect of sensor errors on fighter intercept capability. The primary measure of effectiveness of the fighter defences is the number of interceptions achieved before the raid reaches its target. Errors in the initial GC estimates of raid position, speed and heading are considered. The analysis is concerned primarily with fixed errors, although a procedure is derived which provides the corresponding results for random (normally distributed) errors. Expressions are first derived for the errors which prevent fighter scramble, under each of two scramble policies. The maximum acceptable errors that still enable a scrambled fighter to achieve AI radar detection of the raid are then determined. Finally, if a fighter detects the raid, the feasibility of it correcting its course and intercepting the raid is determined and, if an interception is possible, the final interception point is calculated. The analysis is primarily geometric in nature, and as such requires a pre-defined coordinate system. This could be derived, for example, as in Figure 1.1 of Part 3, in which the y-axis represents a 'target axis', while the fighter base is at the origin of coordinates. We consider a fighter area defence system in which interceptions are regarded as successful only if they are achieved before the raid penetrates the target axis. We refer to the point of intersection of the raid track with the target axis as the raid target.

As in the study of raid indirect routing, just one fighter base is considered, defending a single target against attack from an overflying, concentrated (point) raid. The raid is assumed to fly directly towards its target. Interception is represented by 'collision', the coincidence of the fighter and the raid on their respective tracks, with no allowance made for offsets for forward or rear engagements, or for re-attacks.

2. OPERATIONAL ASSUMPTIONS

Raid Parameters

Table A1 and Figure A1 summarise the raid parameters considered in the analysis. The x- and y-coordinates of the raid at first detection are termed the warning distance and offset respectively. The analysis considers the raid position, actual and estimated, expressed in terms of (x,y) coordinates and in terms of range and bearing from the fighter base. Throughout, the suffixes 't' and 'f' denote actual (true) and estimated (false) raid parameters respectively. Thus the raid is believed to be first detected at the point S_f with coordinates (x_f, y_f), with estimated speed U_f (assumed constant) and estimated heading ψ_f, measured as shown; it is in fact first detected at the point S_t with coordinates (x_t, y_t), while it actually has speed U_t and heading ψ_t.

Fighter Parameters

Given the scramble time and the initial fighter heading, determined by the GC estimates of the raid parameters, three parameters determine the fighters' capabilities - speed V (assumed constant), radar range R and radar angle of scan $\pm\alpha$. The GC estimates of the raid parameters are not updated, and the fighters must themselves detect the raid before they correct their courses (a pessimistic condition). Following initial detection of the raid there is taken to be a fixed delay for sensor and GC processing and transmission of the information, followed by a further reaction delay representing the readiness level of the fighters. It is assumed that during these delays the raid is identified as hostile and approaching in sufficient strength to warrant fighter scramble (subject to the limitations of the scramble policy). There is also a delay between each fighter take-off.

From Figure A1, if a fighter suffers a total delay D between initial detection of the raid by GC sensors and its subsequent scramble, the raid is believed to be at the point S'_f with coordinates (X_f, Y_f) when the fighter takes off, where

$$X_f = x_f - U_f D \cos\psi_f \ ,$$

and (A1)

$$Y_f = y_f + U_f D \sin\psi_f \ .$$

The corresponding actual raid position is at the point S'_t with coordinates (X_t, Y_t), where

$$X_t = x_t - U_t D \cos\psi_t \ ,$$

and (A2)

$$Y_t = y_t + U_t D \sin\psi_t \ .$$

The numerical examples presented in the main text refer only to a raid which is flying at the same constant speed as the fighter and on a direct track to the target which is perpendicular to the target axis, so that

$$V = U_t$$

and (A3)

$$\psi_t = 0 \ .$$

In the analysis which follows of conditions for fighter scramble and detection of the raid, etc., some results are derived only for the special case given by equations (A3). In addition, in order to compare the relative importance of errors in each of the raid parameters, a number of results are presented under the further assumption that just one of the raid parameters has been estimated incorrectly. To this end we define the following error expressions :

$$\mathcal{E}_x = x_f - x_t$$

$$\mathcal{E}_y = y_f - y_t$$

$$\mathcal{E}_U = U_f - U_t \qquad \text{(A4)}$$

$$\mathcal{E}_\psi = \psi_f - \psi_t$$

It is felt that the timeliness of the attack information, i.e. the warning distance achieved, is one of the most crucial fighter parameters. The principal results of this Appendix are illustrated in Figures A4-A7, in which we present, for a fixed fighter take-off delay D, the maximum acceptable error for each of the raid parameters considered separately (i.e. $\mathcal{E}_x$, $\mathcal{E}_y$, $\mathcal{E}_\psi$ and $\mathcal{E}_U$), as a function of the actual warning distance, x_t. Tables 1 and A2 , respectively, summarise the numerical inputs in the examples presented and define the symbols used to distinguish the different components of the graphs. Figures A4-A7 are discussed in more detail later.

3. FIGHTER SCRAMBLE LIMITATIONS

We derive expressions for the maximum acceptable error for a fighter to scramble, with a delay D before take-off, given that Ground Control may estimate incorrectly the raid's position and track (speed and heading) at first detection. Scramble policies I, II and IV, as defined in Part 3, are considered. The general expressions for the maximum error for scramble are obtained from Part 3; we then simplify these by considering in turn errors in only one of the raid parameters, for the special case given by equation (A3). The resultant equations are manipulated to give expressions linking the error - E_x, E_y, E_ψ or E_U - with the actual warning distance x_t.

3.1 Scramble Policy I

Under this policy a fighter is scrambled only if it expects to intercept before the raid penetrates the target axis. From equation (A11) of Part 3 this condition is given by

$$\frac{|y_f + x_f \tan \psi_f|}{V} \leq \frac{x_f \sec \psi_f - U_f D}{U_f} \qquad \text{(A5)}$$

This gives the following expressions for the range of errors under which scramble is permissible under this policy, when each of the raid parameters is considered separately, in the special case given by equations (A3):-

$$E_x \geq |y_t| + VD - x_t \qquad \text{(A6)}$$

$$-x_t + (VD - y_t) \leq E_y \leq x_t - (VD + y_t) \qquad \text{(A7)}$$

$$E_U \leq (V.x_t / (VD + |y_t|)) - V \qquad \text{(A8)}$$

$$VD - x_t \sec E_\psi \leq y_t + x_t \tan E_\psi \leq x_t \sec E_\psi - VD \qquad \text{(A9)}$$

Equation (A9) gives the pair of inequalities

$$(VD - y_t) \cos E_\psi \leq x_t (1 + \sin E_\psi) \qquad \text{(A10)}$$

and

$$(VD + y_t) \cos E_\psi \leq x_t (1 - \sin E_\psi) \qquad \text{(A11)}$$

In the graphs illustrating the limiting conditions under scramble policy I, presented in Figures A4-A7, we do not show, in general, the solutions to inequalities (A6)-(A11) corresponding to unrealistic values of the errors E_x, E_y, E_ψ and E_U.

3.2 Scramble Policy II

Under this policy a fighter is scrambled as long as it possesses a geometrically feasible interception course. From equation (A14) of Part 3 the limiting case is given by

$$X_f = \rho_f \left(\sin\psi_f \pm k_f \cos\psi_f\right) \quad , \qquad (A12)$$

where

$$X_f = x_f - U_f D \cos\psi_f \quad , \qquad (A13)$$

$$k_f^2 = \frac{U_f^2}{V^2} - 1 \quad , \qquad (A14)$$

and ρ_f is the shortest distance from the fighter base to the estimated raid track:

$$\rho_f = x_f \sin\psi_f + y_f \cos\psi_f \quad . \qquad (A15)$$

This gives the following expressions for the maximum acceptable error for scramble, when each of the raid parameters is considered separately, in the special case given by equations (A3).

(i) Error in estimated warning distance, x, only

$$E_x = -x_t + VD \quad . \qquad (A16)$$

(ii) Error in estimated raid offset, y, only

Scramble occurs for all values of E_y provided

$$x_t \geqslant VD \quad . \qquad (A17)$$

(iii) Error in estimated raid speed, U, only

Equation (A14) gives

$$U_f^2\left(y_t^2 - (VD)^2\right) + 2U_f\left(V^2 D x_t\right) - V^2 r_t^2 = 0 \quad , \qquad (A18)$$

where

$$r_t^2 = x_t^2 + y_t^2 \quad .$$

This has the following solutions:

(a) $|y_t| = VD$

$$E_U = \frac{r_t^2}{2x_t D} - U_t \qquad (A19)$$

(b) $|y_t| \neq VD$

$$E_U = \frac{V\{-x_t VD \pm y_t(r_t^2-(VD)^2)^{1/2}\}}{(y_t^2-(VD)^2)} - U_t \quad . \qquad (A20)$$

(iv) Error in estimated raid heading, ψ, only

$$x_t = VD \sec E_\psi + y_t \tan E_\psi \qquad (A21)$$

3.3 Scramble policy IV

Under this policy a fighter is scrambled only if it is expected to intercept within a specified time say $\bar{T}$, after take-off. From equation (A26) of Part 3 the limiting condition for interception is given by

$$x_f \cos \psi_f - y_f \sin \psi_f = U_f(D+\bar{T}) - ((V\bar{T})^2 - p_f^2)^{1/2} \qquad (A22)$$

This gives the following expressions for the maximum error in each of the raid parameters x, y and U for scramble in the special case given by equations (A3); the expression for E_ψ may also be obtained with a little manipulation.

$$E_x = -x_t + V(D+\bar{T}) - ((V\bar{T})^2 - y_t^2)^{1/2} \qquad (A23)$$

$$x_t = V(D+\bar{T}) - ((V\bar{T})^2 - (E_y + y_t)^2)^{1/2} \qquad (A24)$$

$$E_U = \frac{x_t + ((V\bar{T})^2 - y_t^2)^{1/2}}{(D+\bar{T})} - U_t \qquad (A25)$$

4. FIGHTER HEADING

Given that a fighter has scrambled onto an expected interception course under policy I, II or IV, its initial heading φ , measured from the positive x-axis (see Figure A1), is given by equation (A10) of Part 3,

$$\sin\varphi = \frac{X_f \cdot \rho_f \cdot \frac{U_f}{V} + Y_f\left(R_f^2 - \rho_f^2 \frac{U_f^2}{V^2}\right)^{1/2}}{R_f^2} , \qquad (A25)$$

where R_f is the estimated distance of the raid from the fighter base at the moment the fighter takes-off;

$$R_f^2 = X_f^2 + Y_f^2 .$$

If we assume that the estimated raid speed equals the fighter speed, so that

$$V = U_f , \qquad (A26)$$

then we easily obtain the following useful expression for the half-angle $\frac{\varphi}{2}$:

$$\tan\frac{\varphi}{2} = \frac{\tan\frac{\psi_f}{2} + \frac{Y_f}{X_f}}{1 - \tan\frac{\psi_f}{2} \cdot \frac{Y_f}{X_f}} \qquad (A27)$$

5. RAID VELOCITY RELATIVE TO THE FIGHTER

As might be expected, the analysis is simplified if we regard the fighter as 'reduced to rest' at the fighter base at the moment it takes off, by considering the velocity $\underline{W}$ of the raid relative to the fighter. If the raid has relative speed W and heading ω as shown in Figure A2(a,b), then W and ω are given by

$$W\sin\omega = V\sin\phi - U_t \sin\psi_t ,$$
$$W\cos\omega = V\cos\phi + U_t \cos\psi_t , \qquad \text{(A28)}$$

so that

$$W^2 = V^2 + U_t^{\,2} + 2VU_t \cos(\phi+\psi_t)$$

and

$$\tan\omega = \frac{V\sin\phi - U_t \sin\psi_t}{V\cos\phi + U_t\cos\psi_t} . \qquad \text{(A29)}$$

Figure A2(a,b) also shows for later use, $-\underline{W}$, the fighter velocity relative to the raid.

5.1 Example

In the standard example given by equations (A3), in which the raid attacks the target along a track perpendicular to the target axis and the fighter speed equals the raid speed, then

$$\omega = \frac{\phi}{2}$$

and (A30)

$$W = 2V\cos\frac{\phi}{2} .$$

6. FIGHTER DETECTION OF THE RAID

Given that a fighter has scrambled to attempt an interception, we now determine the conditions under which it can detect the raid while on its initial interception course. The fighter is assumed to have a deterministic detection sector, with radar range R and angle of scan $\pm\alpha$ $(0 \leq \alpha \leq \pi)$. Within this sector a fighter detects with 100% probability, while it has zero probability of detection ouside it. Furthermore, we adopt a geometrical analysis only, and do not consider the problems of detecting targets with relative velocities such that they are in high clutter regions when the fighter's radar is in pulse doppler mode. The fighter model could be used if, instead of a broad and relatively straightforward geometrical approach, a detailed model of the fighters' radar detection capability and clutter rejection ability was required.

As shown in Figure A2(a,b), the angle Φ is defined by

$$\Phi = \phi - \omega \quad . \qquad \text{(A31)}$$

Thus Φ is the angle made by the fighter track relative to the raid with the initial fighter heading. If the angle ξ is defined as shown in Figure A2(a,b), then we define β by

$$\beta = \xi + \Phi \quad . \qquad \text{(A32)}$$

Hence β denotes the angle made by the raid with the fighter track when it first comes within radar range R; our convention is that β is positive if the fighter track passes ahead of the raid and negative if it passes behind the raid. If we define p_r to be the shortest distance from the fighter base (at the origin of coordinates) to the raid track $\underline{W}$ relative to the fighter, then

$$p_r = \dot{X}_t \sin\omega - Y_t \cos\omega \;, \qquad \text{(A33)}$$

and ξ is given by

$$p_r = R \sin\xi \quad . \qquad \text{(A34)}$$

Annex C investigates the conditions on α and Φ necessary for detection; these may be summarised as follows.

(a) $0 \leq \alpha \leq \frac{\pi}{2}$

The fighter eventually detects the raid if and only if

$$-\min\left(\alpha, \frac{\pi}{2} - \Phi\right) \leq \beta \leq \max(\alpha, \Phi) \quad . \qquad \text{(A35)}$$

(b) $\frac{\pi}{2} \leqslant \alpha \leqslant \pi$

The fighter eventually detects the raid if and only if

$$-\min\left(\alpha, \frac{\pi}{2} - \Phi\right) = -\left(\frac{\pi}{2} - \Phi\right) \leqslant \beta \leqslant \min\left(\frac{\pi}{2} + \Phi, \alpha\right) \quad . \qquad (A36)$$

The criteria (A35) and (A36) for fighter detection of the raid will now be considered in more detail. For simplicity we restrict attention to the case $0 \leqslant \alpha \leqslant \pi/2$, although the analysis can easily be extended to the case $\pi/2 \leqslant \alpha \leqslant \pi$.

Example

In the standard example given by equations (A3), substituting from equation (A30) into (A31) and (A32) gives

$$\Phi = \frac{\varphi}{2} \qquad (A37)$$

and

$$\beta = \xi + \frac{\varphi}{2} \quad . \qquad (A38)$$

6.1 Detection at limits of fighter radar scan ($\beta = \pm\alpha$)

If detection occurs at the limit of the fighter radar scan then either $\beta = +\alpha$ or $\beta = -\alpha$, depending on whether the fighter passes ahead of or behind the raid respectively. Note that from (A35) the limiting condition for detection when the fighter passes ahead of the raid is given by $\beta = +\alpha$ only if

$$\alpha \geq \Phi \tag{A39}$$

Similarly, the limiting condition for detection if the fighter passes behind the raid is given by $\beta = -\alpha$ only if

$$\alpha \leq \frac{\pi}{2} - \Phi \tag{A40}$$

From equation (A32), if $\beta = \pm\alpha$ we have

$$\xi = \pm\alpha - \Phi \tag{A41}$$

Substituting from (A33) and (A34) gives

$$X_t \sin\omega - Y_t \cos\omega = R \sin(\pm\alpha - \Phi) \tag{A42}$$

Equations (A25), (A29), (A31) and (A42) may be solved numerically to give the relationships between the maximum acceptable error for fighter detection, in the estimate of each of the raid parameters, and the actual warning distance x_t at which the raid is first detected by GC sensors.

Examples

(i) In the particular case given by (A3), equations (A39) and (A40) reduce respectively to

$$\frac{\phi}{2} \leq \alpha \tag{A43}$$

and

$$\frac{\phi}{2} \leq \frac{\pi}{2} - \alpha \tag{A44}$$

Equation (A42) easily reduces to

$$\tan\frac{\phi}{2} = \frac{Y_t \pm R\sin\alpha}{X_t + R\cos\alpha} \tag{A45}$$

where

$$Y_t = y_t$$

and

$$X_t = x_t - VD$$

The simplified expression (A45) may again be solved numerically with (A25) to give the relationships between the maximum allowable errors for detection - E_x, E_y, E_ψ and E_U - and the actual warning distance x_t.

(ii) As a further illustration, consider the additional assumption that only the initial estimate of the raid position is in error, as shown in Figures A3 and A4. Hence

$$E_\psi = \psi_f - \psi_t = \psi_f - 0 = 0$$

and (A46)

$$E_U = U_f - U_t = U_f - V = 0 \ .$$

Substituting into equation (A27) then gives the following simple expression for $\phi/2$:

$$\tan\frac{\phi}{2} = \frac{y_f}{x_f - VD} \ . \qquad \text{(A47)}$$

Finally, by substituting from (A47) into (A45) we obtain the following expressions if we consider the errors E_x, E_y in the raid warning distance and offset separately :

(a) Warning distance only in error

$$E_x(y_t \pm R\sin\alpha) = R(y_t\cos\alpha \pm VD\sin\alpha) \mp R\sin\alpha . x_t \ , \qquad \text{(A48)}$$

(b) Raid offset only in error

$$E_y(x_t - VD + R\cos\alpha) = \pm R\sin\alpha(x_t - VD) - R\cos\alpha . y_t \ . \qquad \text{(A49)}$$

6.2 Limit on detection as fighter passes ahead of the raid ($\beta = \Phi$)

From (A35), if the fighter passes ahead of the raid and $\alpha < \Phi$, the limiting condition for detection is given by

$$\beta = \Phi \quad . \tag{A50}$$

From equation (A32) this is equivalent to

$$\xi = 0 \quad . \tag{A51}$$

In this limiting case the errors E_x, E_y, E_U, E_ψ in the estimates of the raid parameters combine so that the fighter is actually on a valid interception course. Note that the actual interception point will in general be different from the expected interception point. From (A33) and (A34) equation (A51) gives

$$\tan \omega = \frac{Y_t}{X_t} \quad . \tag{A52}$$

Example

If equations (A3) hold and if also only the initial estimate of the raid position is in error, then substituting from (A47) and (A30) into (A52) gives

$$\tan \frac{\phi}{2} = \frac{y_t}{x_t - VD} = \frac{y_F}{x_F - VD} \quad , \tag{A53}$$

i.e.

$$E_x \cdot y_t = E_y \cdot (x_t - VD) \quad . \tag{A54}$$

In the numerical examples presented later we evaluate the maximum allowable error for successful interception, when each of the raid parameters is considered separately; equation (A51) then corresponds simply to zero error in the estimate of the appropriate raid parameter.

6.3 Limit on detection as fighter passes behind the raid $(\beta = -(\frac{\pi}{2} - \Phi))$

If the fighter passes behind the raid and $\pi/2 - \Phi < \alpha$, the limiting condition for detection is given, from (A35), by

$$\beta = -(\frac{\pi}{2} - \Phi) \qquad \text{(A55)}$$

From equation (A32) this reduces to

$$\xi = -\frac{\pi}{2} \qquad \text{(A56)}$$

Substituting into (A34) gives

$$p_r = -R \qquad \text{(A57)}$$

As can be seen from Figures C2 of Annex C, in this case the limiting detection occurs when the relative raid track is tangential to the fighter's radar detection sector. In the general case equation (A57) is solved by substituting from (A33), (A28) and (A25) to give an expression linking the maximum acceptable errors in the initial estimates of the raid parameters for detection, with the warning distance at which the raid is first detected by Ground Control sensors.

In the particular case defined by equations (A3) and considering positional errors only, the solution of (A57) simplifies considerably. We find from (A30), (A33) and (A47) that

$$p_r = \frac{y_f E_x - (x_f - VD) E_y}{((x_f - VD)^2 + y_f^2)^{1/2}} \qquad \text{(A58)}$$

This gives the following expressions.

(i) Error in estimated warning distance, x, only

$$x_t = (VD - E_x) \pm \frac{y_t}{R}(E_x^2 - R^2)^{1/2} \quad (E_x \geq R) \qquad \text{(A59)}$$

(ii) Error in estimated raid offset, y, only

$$x_t = VD \mp R(E_y + y_t)/(E_y^2 - R^2)^{1/2} \quad (E_y \geq R) \qquad \text{(A60)}$$

In equations (A59) and (A60) we take the top-most sign. The other sign also gives a solution for 'detection' at a tangent to the fighter's arc of radar cover (extended in azimuth if necessary); in this case it corresponds to the fighter passing ahead of the raid and if $0 \leq \alpha \leq \frac{\pi}{2}$ does not give a valid solution.

7. FINAL INTERCEPTION POINT

Given that a fighter has scrambled in response to a threat warning and has detected the enemy raid, we consider finally the maximum acceptable errors such that the fighter can successfully intercept the raid as a result of this detection. We restrict ourselves for simplicity to the particular case given by equation (A3), on which Figures A4-A7 are based. We use the general interception algorithm developed in Part 3, assuming that fighters can respond immediately to the track-change and that interception courses are calculated purely on their geometrical feasibility, ignoring fuel limitations. In fact in realistic scenarios the limitations on fighter intercept capability consequent upon detection, for a raid to be intercepted on or before the target axis, are generally independent of the fighter endurance. If D_c denotes the fighter reaction delay after a detection, then to simplify the computation we assume that

$$D_c = 0 \quad . \qquad \text{(A61)}$$

Figures A3(a,b) illustrate the raid and fighter geometry under consideration, for the cases in which the fighter passes ahead of and behind the raid respectively. Here F_d denotes the fighter position when it detects the raid and R_d denotes the corresponding raid position. We concentrate on the conditions for successful interception of the raid, before it reaches the target axis; for completeness we first calculate the conditions under which the fighter possesses an interception course, regardless of the final interception point.

7.1 Geometrically feasible interception course

If the angle α is defined as shown in Figures A3(a) and A3(b), we may write respectively

$$\begin{aligned} \phi &= |\beta| + \alpha \\ &= \beta + \alpha \end{aligned} \qquad \text{(A62)}$$

and

$$\begin{aligned} \alpha &= |\beta| + \phi \\ &= -\beta + \phi \quad . \end{aligned} \qquad \text{(A63)}$$

Hence, taking $\alpha = \frac{\pi}{2}$ and substituting for β from equation (A38), we find from (A62) and (A63) that

$$\zeta = \frac{\phi}{2} - \frac{\pi}{2} \quad . \qquad \text{(A64)}$$

Substituting from (A33) and (A34) gives

$$\tan\frac{\phi}{2} = \frac{Y_t - R}{X_t} \quad . \qquad \text{(A65)}$$

Equation (A65) may be solved numerically to give the maximum errors E_x, E_y, E_ψ and E_U for a geometrically feasible interception course, as a function of the warning distance x_t. Similarly (although this case has little practical significance), if in (A62) and (A63) we take $\alpha = -\frac{\pi}{2}$ we find

$$\zeta = \frac{\phi}{2} + \frac{\pi}{2} \qquad \text{(A66)}$$

from which

$$\tan\frac{\phi}{2} = \frac{Y_t + R}{X_t} \qquad \text{(A67)}$$

<u>Example</u>

If we consider positional errors only in the GC initial estimate of the raid track, so that $E_U = E_\psi = 0$, then from equation (A47)

$$\tan\frac{\phi}{2} = \frac{Y_f}{X_f} = \frac{y_f}{x_f - VD}$$

Substituting into (A65) gives

$$E_x(y_t - R) - RX_t = E_y X_t \qquad \text{(A68)}$$

Similarly, substituting into (A67) gives

$$E_x(y_t + R) + RX_t = E_y X_t \qquad \text{(A69)}$$

Finally, if we consider the errors in the initial estimates of the raid's x- and y- coordinates separately, equations (A68) and (A69) reduce as follows.

(i) <u>$E_y = 0$</u>

$$E_x = \frac{\pm R\,(x_t - VD)}{y_t \mp R} \qquad \text{(A70)}$$

(ii) <u>$E_x = 0$</u>

$$E_y = \mp R \qquad \text{(A71)}$$

7.2 Interception before the raid reaches the target

From Figure A3(a,b) it may be seen that, under the assumptions given by equations (A3), the coordinates of the final interception point may be calculated directly from the relative fighter and raid positions when the fighter detects the raid - shown as the points F_r and S_t' respectively. If X denotes the x-coordinate of the final interception point relative to the point F_r, then the actual x-coordinate X_a of the interception point is simply given by

$$X_a = X + x_d \quad . \qquad \text{(A72)}$$

Here x_d is the x-coordinate of the point F_d, the actual position of the fighter at detection. Expressions for X and x_d are derived below. An interception is deemed to be successful if it occurs before the raid reaches the target axis, i.e., if

$$X_a \geqslant 0 \quad . \qquad \text{(A73)}$$

If the point F_r has coordinates (x_r, y_r), then from Figure A3(a,b)

$$R^2 = (X_t - x_r)^2 + (Y_t - y_r)^2 \qquad \text{(A74)}$$

and

$$y_r = x_r . \tan \frac{\phi}{2} \quad . \qquad \text{(A75)}$$

Substituting from (A75) and (A33) into (A74) leads to

$$x_r = \cos^2 \frac{\phi}{2} \left\{ X_t + Y_t \tan \frac{\phi}{2} - \sec \frac{\phi}{2} (R^2 - \rho_r^2)^{1/2} \right\} \quad . \qquad \text{(A76)}$$

Now, if T is the time taken for the fighter to reach the point F_r on its relative track, then

$$T = \frac{x_r \sec \phi/2}{W} \quad , \qquad \text{(A77)}$$

where W is the fighter speed relative to the raid; from (A30) this reduces to

$$T = \frac{x_r}{2V \cos^2 \frac{\phi}{2}} \quad . \qquad \text{(A78)}$$

Hence the coordinates (x_d, y_d) of the point F_d are given by

$$x_d = VT \cos \phi = \frac{1}{2} \cos \phi \left\{ X_t + Y_t \tan \frac{\phi}{2} - \sec \frac{\phi}{2} (R^2 - \rho_r^2)^{1/2} \right\} \qquad \text{(A79)}$$

and similarly

$$y_d = VT \sin \phi = y_r \quad . \qquad \text{(A80)}$$

Finally, knowing the coordinates (x_r, y_r) of the point F_r, the coordinates (X,Y) of the final interception point relative to F_r and S_t' may be derived from equation (A5) of Part 3; this gives

$$X = \frac{(X_t - x_r)^2 - (Y_t - y_r)^2}{2(X_t - x_r)}$$

and (A81)

$$Y = Y_t - y_r \quad .$$

Hence the coordinates (X_a, Y_a) of the interception point relative to the origin are given by

$$X_a = X + x_d$$

and of course

$$Y_a = Y + y_d = Y_t - y_r + y_d = Y_t = y_t \quad .$$

8. EXAMPLES

8.1 Numerical Inputs

The numerical inputs used in the specific examples presented are summarised in Table 1 (p.7). The fighter base is taken to be Coningsby, with the target at the maximum offset between Coningsby and Leuchars, i.e. 108 nm. along the target axis. The fighters and raid travel at equal speed and the raid is actually flying along a direct track perpendicular to the target axis, so that

$$V=U_t$$

and

$$\psi_t=0 \quad .$$

8.2 Results

The results are presented in Figures A4-A7, while Table A2 contains a key to the notation. The maximum acceptable error is plotted as a function of the actual warning distance of the raid at first GC sensor detection. Results are presented for errors in the GC estimates of the raid's warning distance (x-coordinate), offset (y-coordinate), heading and speed, each considered separately. A delay of 10 minutes is assumed between first detection of the raid and fighter scramble; from Table 1 it can be seen that these figures therefore refer to the first fighter scrambled. We suppose an expression for the total delay before take-off, D, for the I-th fighter scrambled given by

$$D=(D_p+D_r) + I.D_s \quad . \tag{A82}$$

Here D_p and D_r represent the GC processing and transmission time and the fighters' readiness level respectively, while D_s is a delay between successive fighter scrambles.

The expressions for the maximum acceptable errors such that the fighter can still detect the raid, and then successfully intercept it, depend in a complex manner on the fighter take-off delay D. Note that the technique adopted in Part 3 does not apply here; this would correspond to finding the maximum take-off delay for interception (or detection, or scramble) for given error or errors, then determining the corresponding number of interceptions from equation (A82). This is not valid, for the possibility exists of a penalty for premature scramble - if a specified fighter can intercept, it does not follow that fighters scrambled earlier can also intercept. It is impractical to perform the complete analysis described in this Appendix for each fighter scrambled, so the Fighter Model (see Part 2) was used in order to express the consequences of sensor errors in terms of the measure of effectiveness adopted in the main text, viz. the number of fighters which achieve interceptions before the raid reaches the target axis.

The essential conclusions from Figures A4-A7 are quite simple. With reasonable warning distance realistic systematic errors, considered independently, in the initial GC estimates of the raid parameters have no appreciable effect on fighter intercept capability. Figure A4 shows that an overestimate of the distance of the raid at first detection is less serious than a corresponding underestimate, while Figure A7 shows that very large underestimates in the raid speed can be tolerated.

TABLE A1

PARAMETERS INVOLVED IN THE ANALYSIS

RAID PARAMETERS AT INITIAL DETECTION

	Actual	Estimated by Ground Control (GC)
x-coordinate(warning distance)	x_t	x_f
y-coordinate (offset)	y_t	y_f
range from fighter base	r_t	r_f
bearing " " "	θ_t	θ_f
heading	ψ_t	ψ_f
speed (constant)	U_t	U_f

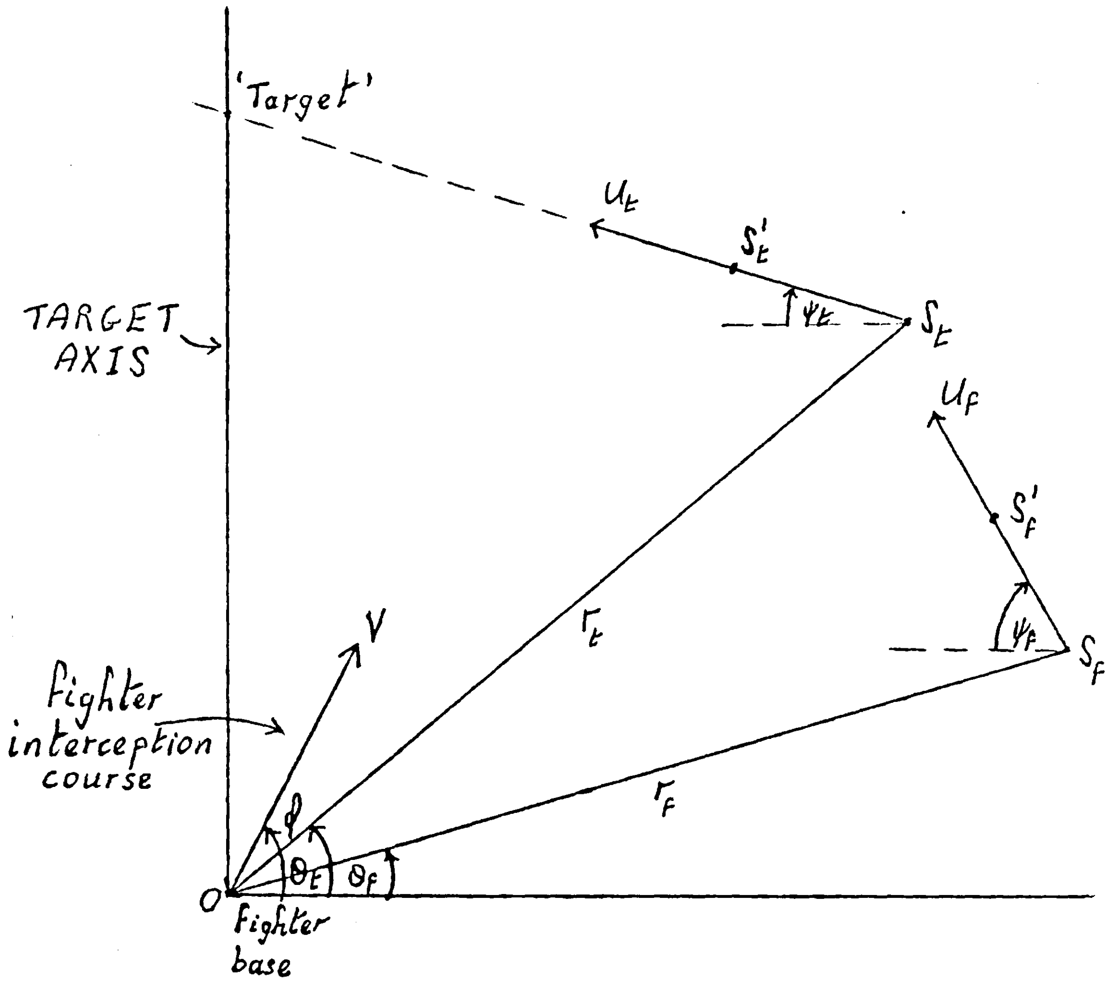
FIGHTER PARAMETERS

speed (constant)	V
delay before take-off	D
radar range	R
radar angle of scan	$\pm\alpha$

TABLE A2

KEY TO FIGURES A4-A7

Symbol	Meaning
S_I	Maximum error under which fighters are scrambled (Policy I)
S_{II}	" " " " " " " (Policy II)
D_α	" " " " " can detect the raid ($\beta = \alpha$)
$D_{-\alpha}$	" " " " " " " " " ($\beta = -\alpha$)
D_Φ	" " " " " " " " " ($\beta = \Phi$)
$D_{-(\frac{\pi}{2}-\Phi)}$	" " " " " " " " " ($\beta = -(\frac{\pi}{2}-\Phi)$)
K_α	graph of $\frac{\Phi}{2} = \alpha$
$K_{(\frac{\pi}{2}-\alpha)}$	graph of $\frac{\Phi}{2} = \frac{\pi}{2} - \alpha$
I_T	maximum error such that the raid can still be intercepted before it reaches the target axis



S_t = actual raid position at first detection by GC

S_f = estimated " " " " " " "

S_t' = actual " " " fighter scramble

S_f' = estimated " " " " "

Figure A1. Fighter and Raid Parameters

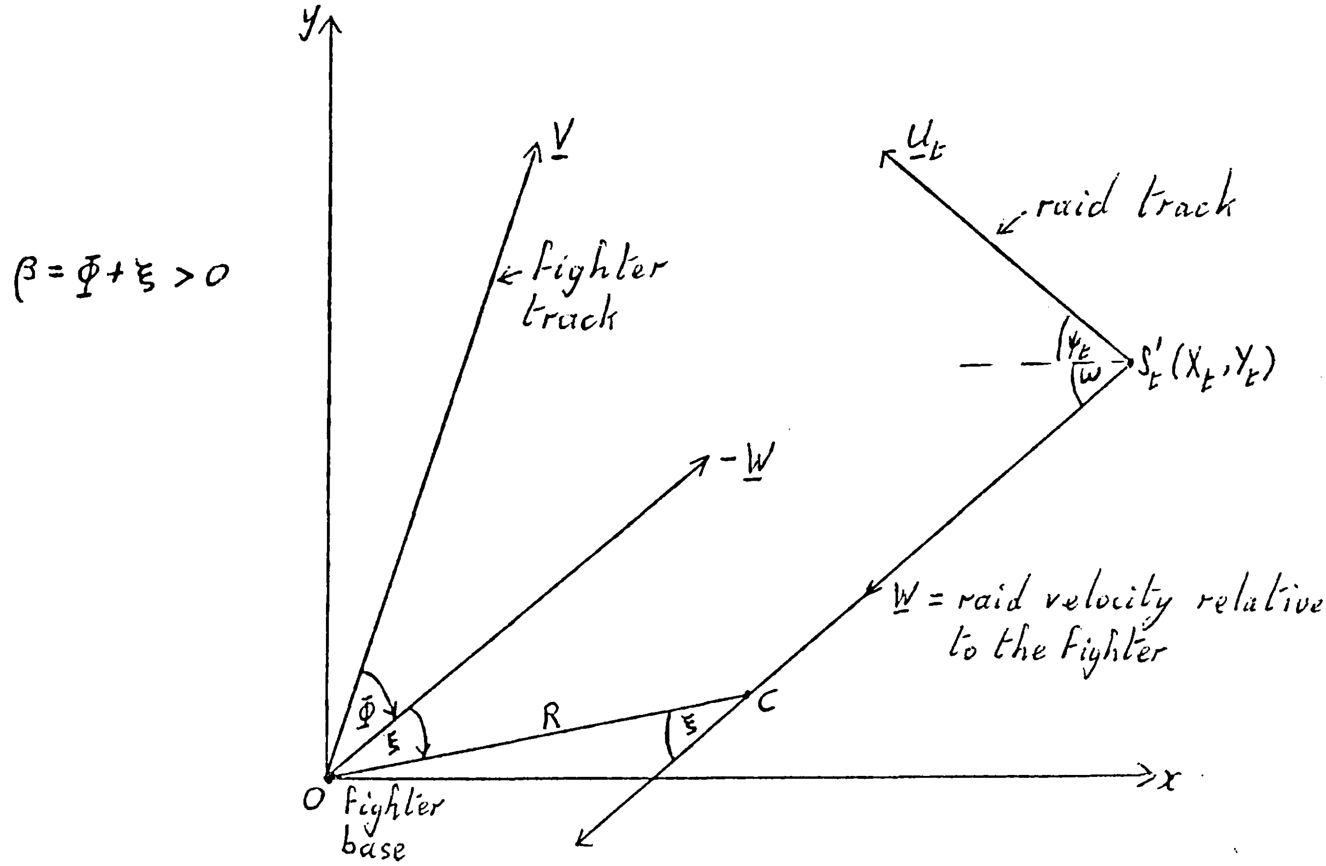


Figure A2(a). Relative raid velocity: fighter passes ahead of the raid

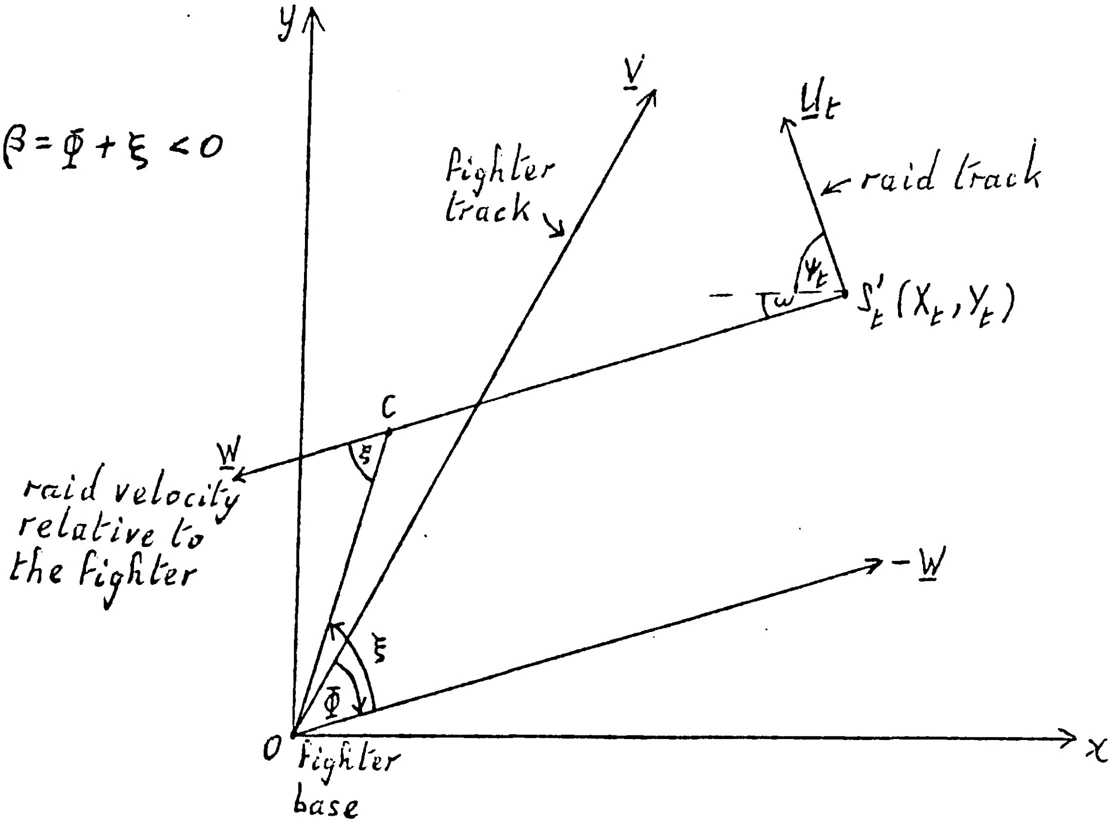


Figure A2(b). Relative raid velocity: fighter passes behind the raid

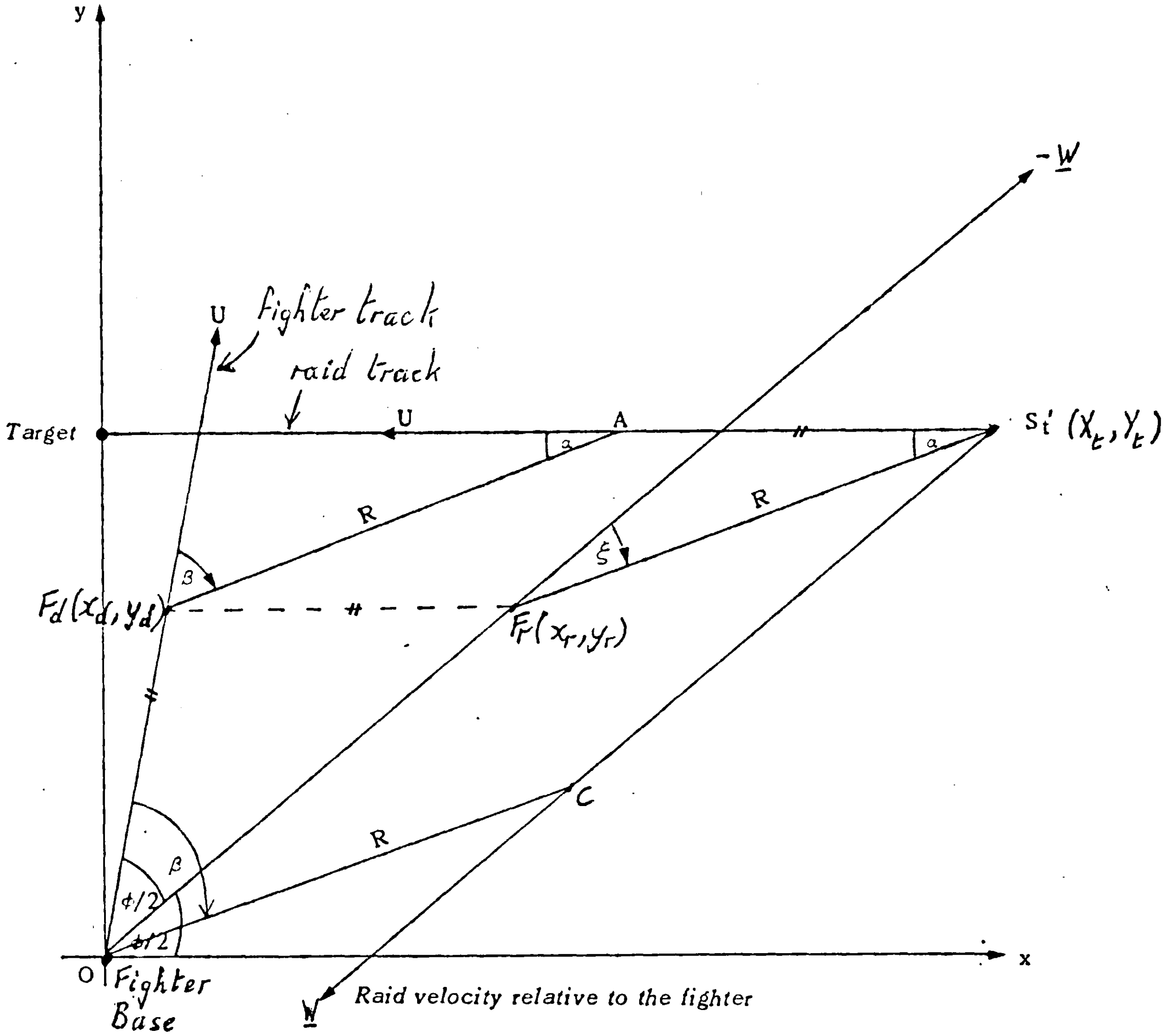


FIGURE A3(a) FIGHTER PASSES AHEAD OF THE RAID

Notes

(i) $|OF_d| = |F_dF_r| = |AS_t'|$

(ii) $\beta = \xi + \frac{\phi}{2}$

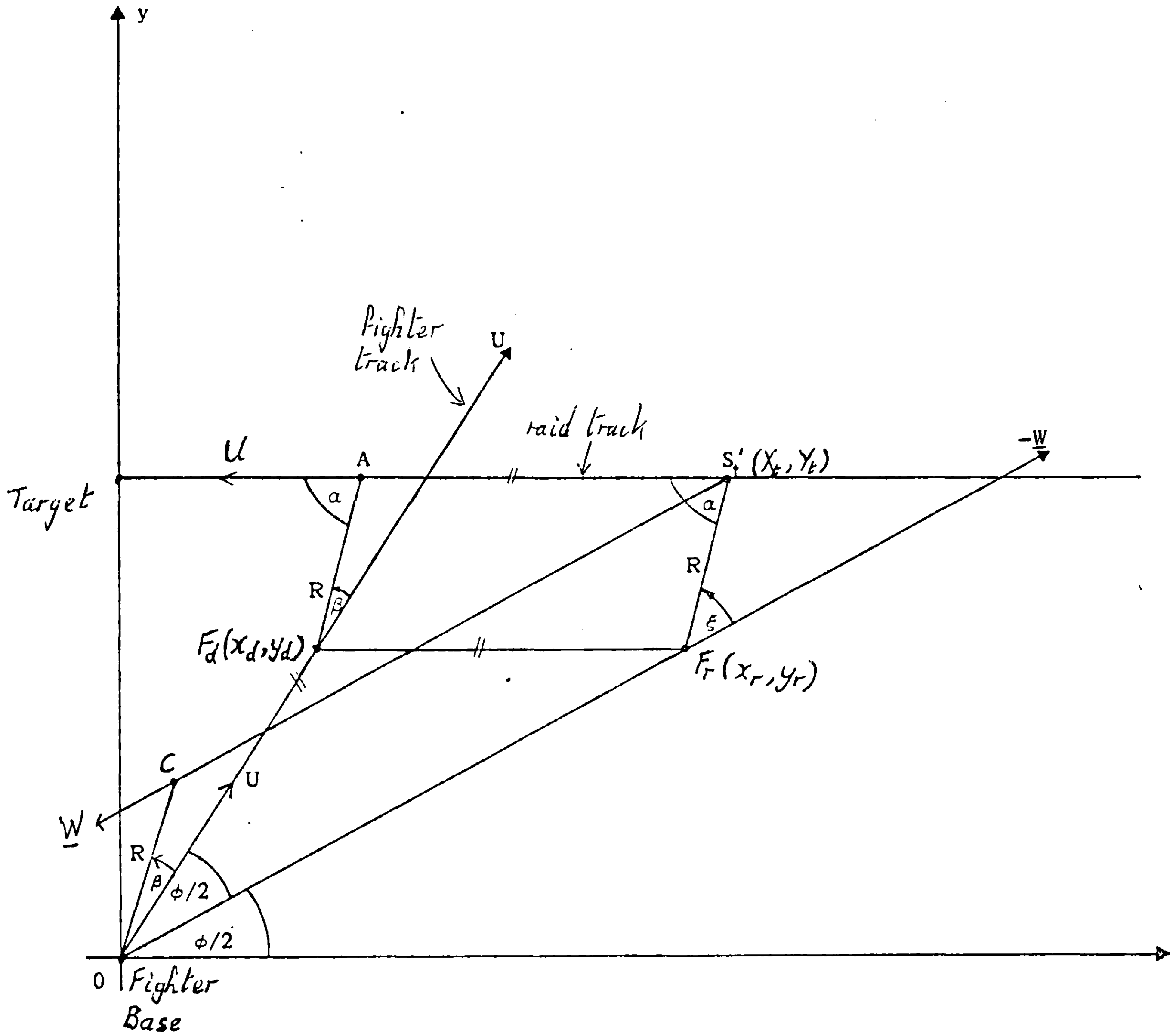


FIGURE A3(b) FIGHTER PASSES BEHIND THE RAID

Notes

(i) $|OF_d| = |F_d F_r| = |AS'_t|$

(ii) $\beta = \xi + \frac{\phi}{2}$ (β, ξ are both negative)

(iii) $\underline{W}$ = raid velocity relative to the fighter

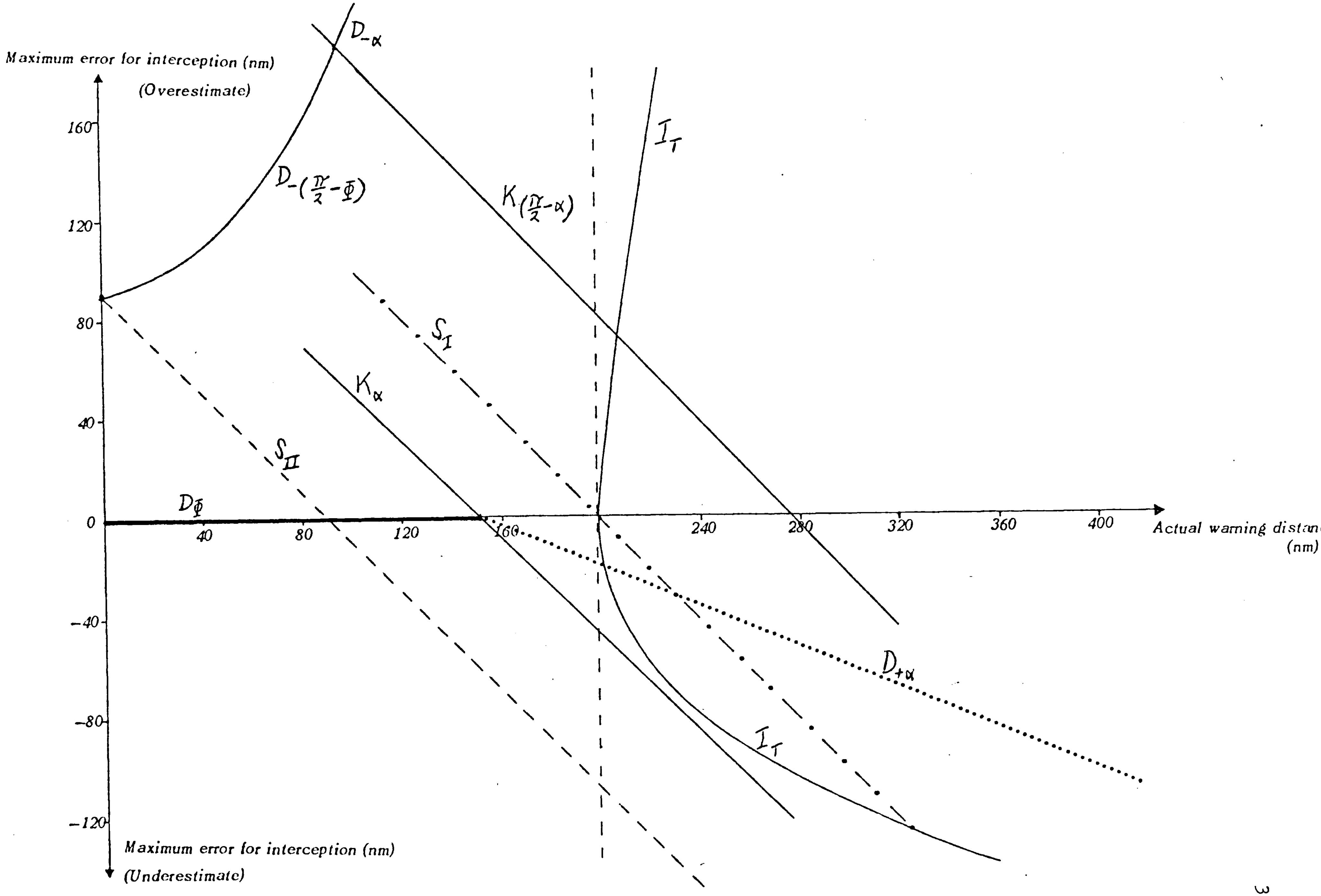


FIGURE A4 MAXIMUM ERROR IN THE ESTIMATED WARNING DISTANCE OF THE RAID AT FIRST DETECTION

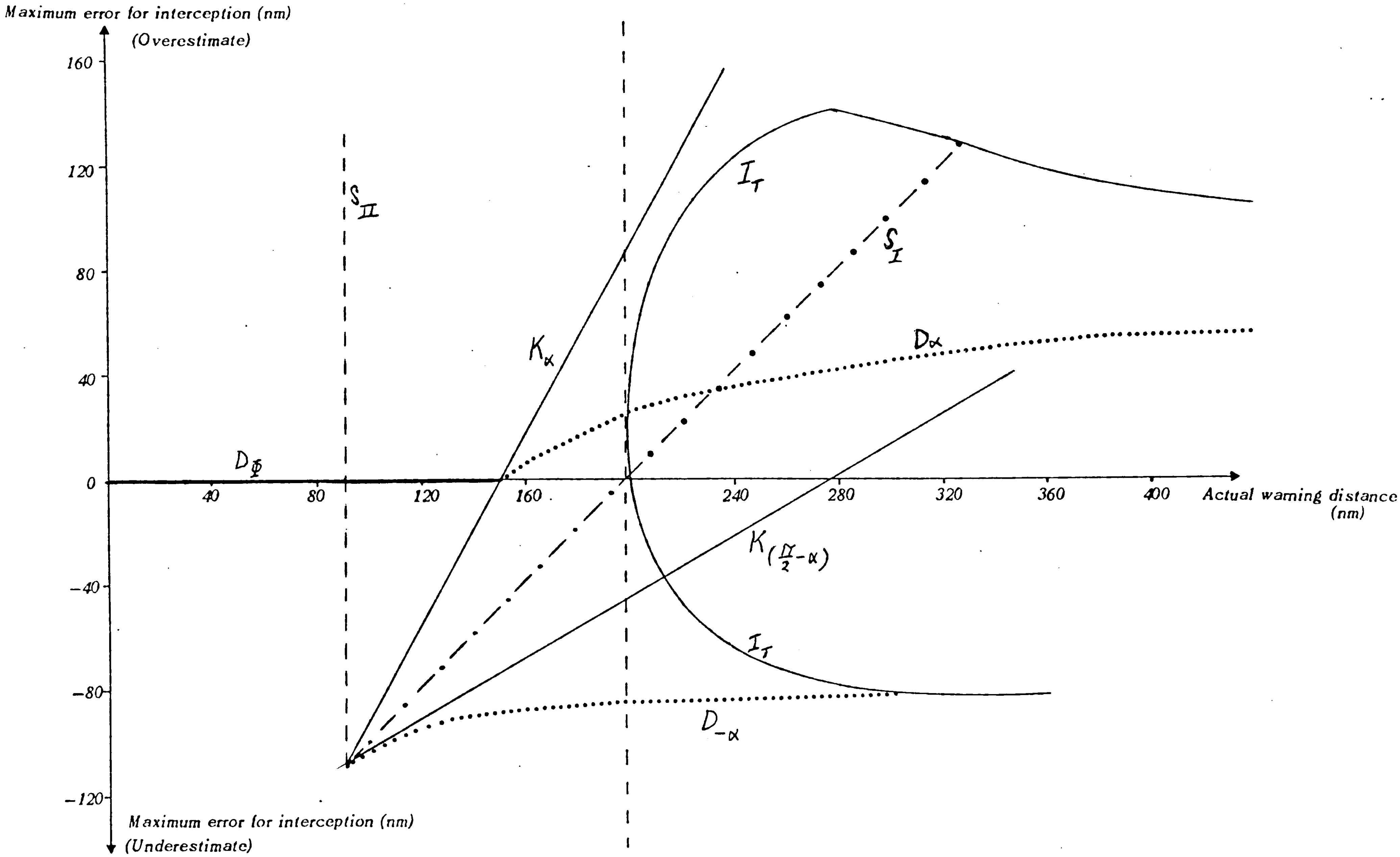


FIGURE A5 MAXIMUM ERROR IN THE ESTIMATED RAID OFFSET AT FIRST DETECTION

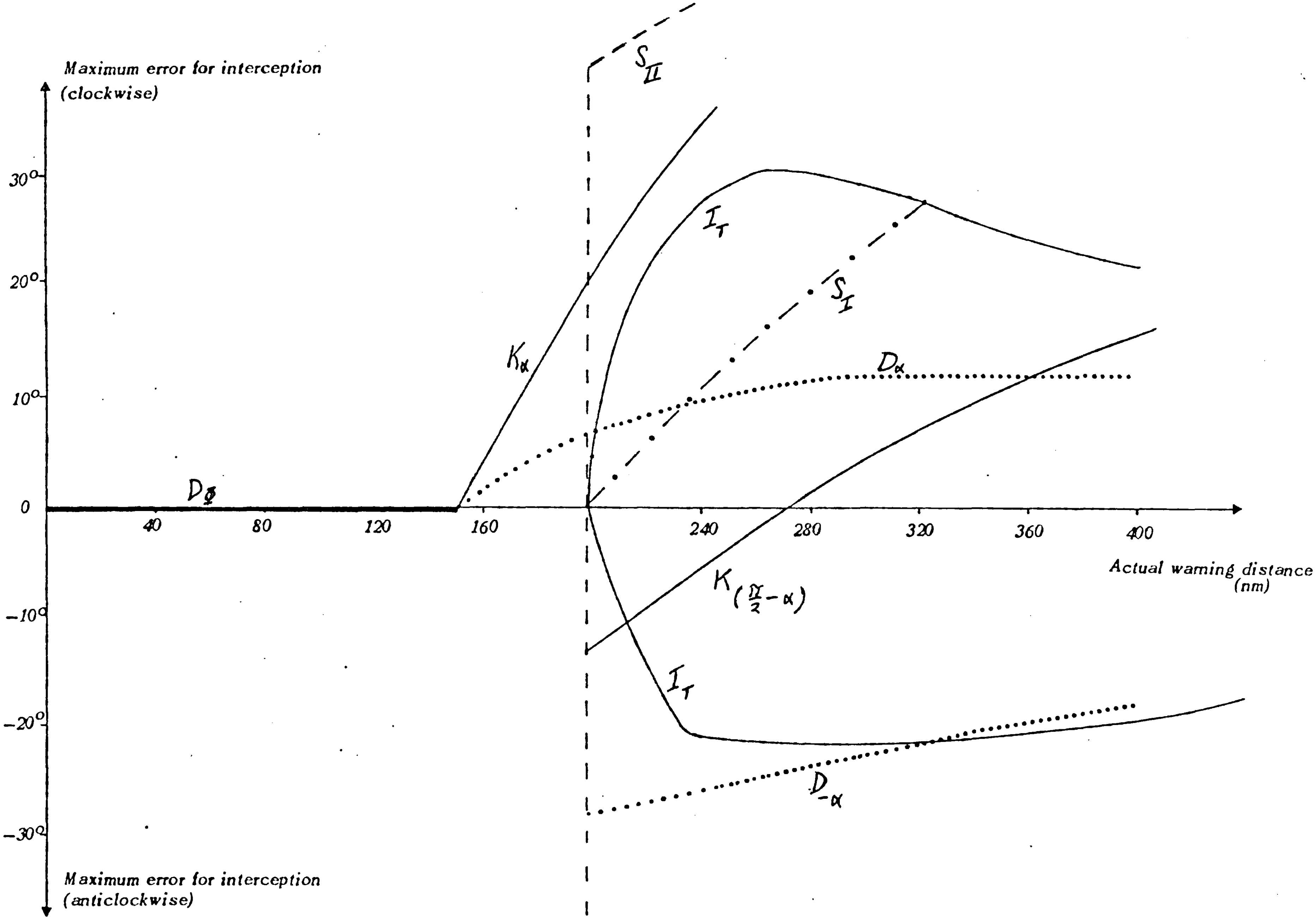


FIGURE A6 MAXIMUM ERROR IN THE ESTIMATED RAID HEADING AT FIRST DETECTION

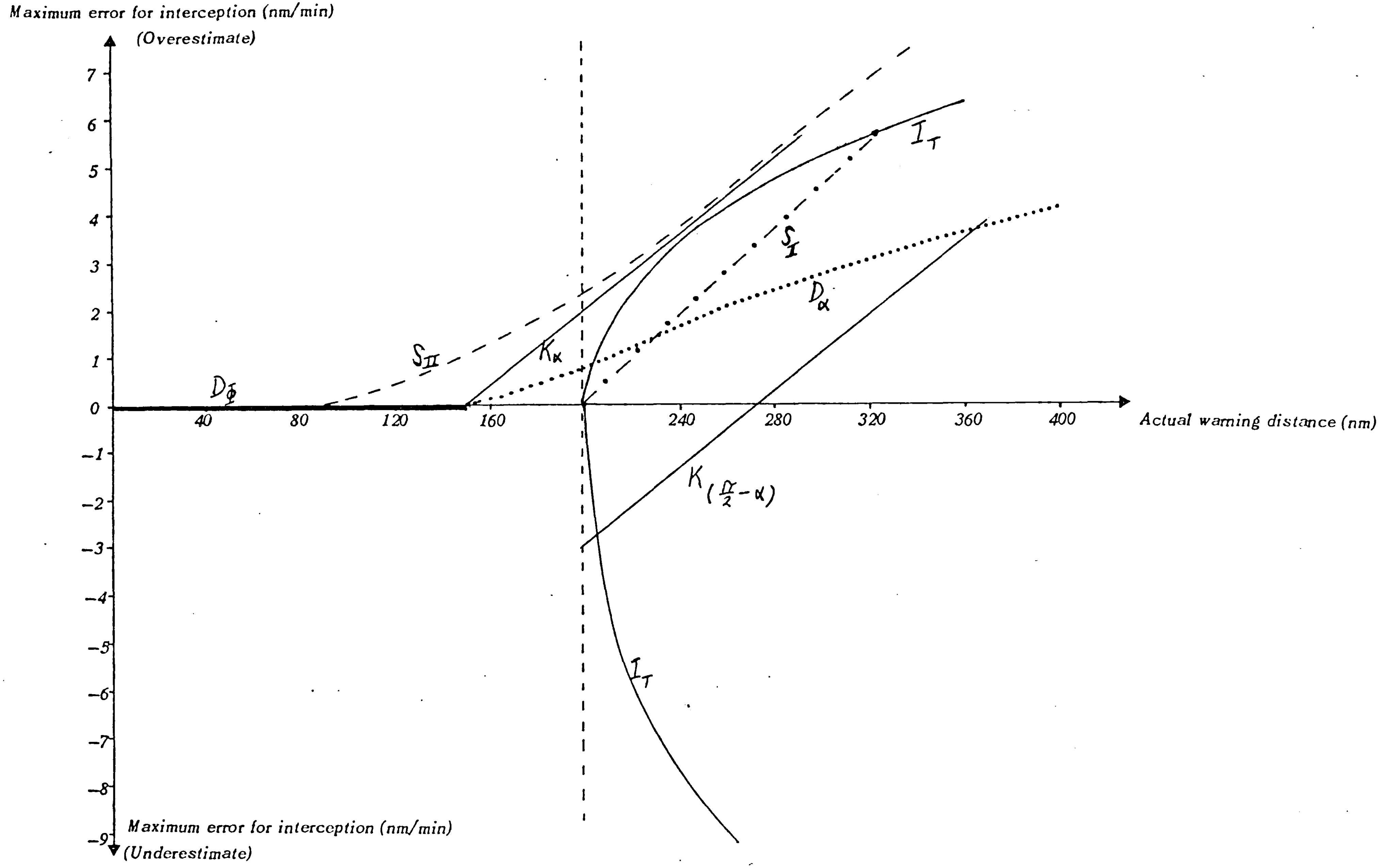


FIGURE A7 MAXIMUM ERROR IN THE ESTIMATED RAID SPEED AT FIRST DETECTION

APPENDIX B

FIGHTER MODEL RESULTS

The numerical inputs used in the fighter model runs are as shown in Table 1 of the main text, except that in the examples presented a fixed actual warning distance of 360nm was taken. Also, the fighter model - a computer simulation - includes a factor which is difficult to study analytically. If in the model a fighter reaches its expected interception point without achieving a detection, it does not continue along its assigned course indefinitely but turns and heads along the positive x-axis towards a holding CAP. It thus flies towards the expected direction of the enemy attack and generally does better than if it simply continues along its initial track.

The model results are presented in Figures B1-B4. They reinforce the basic conclusions of Appendix A, viz with reasonable warning distance practical systematic errors, considered independently, in the initial GC estimates of the raid parameters have no appreciable effect on fighter intercept capability.

Systematic Error to Random Error Conversion

Systematic sensor errors are of less practical relevance than random errors. If the distributions of the errors are known, it is a straightforward calculation to derive the expected number of interceptions from the corresponding results for systematic errors. Random errors, in the numerical examples presented, are assumed to be normally distributed, with zero mean, while a range of variances is considered. The technique is applied to the fighter model results illustrated in Figures B1-B4, to give the expected number of interceptions as a function of the standard deviation of the normal error in each of the raid parameters. These results are presented in Figures 5-8 of the main text.

Suppose I(x) denotes the number of interceptions achieved (out of a maximum of 20) with a systematic error, x, in one of the raid parameters - warning distance, offset, heading or speed. If ϕ denotes the probability density function of a normal random variable with zero mean and variance σ^2, the expected number of interceptions, $E(\sigma)$, is given by

$$E(\sigma) = \int_{-\infty}^{\infty} I(x).\phi(x)\, dx \qquad \text{(B1)}$$

The integral in equation (B1) is calculated numerically to any given accuracy, with values of I(x) obtained from Figures B1-B4.

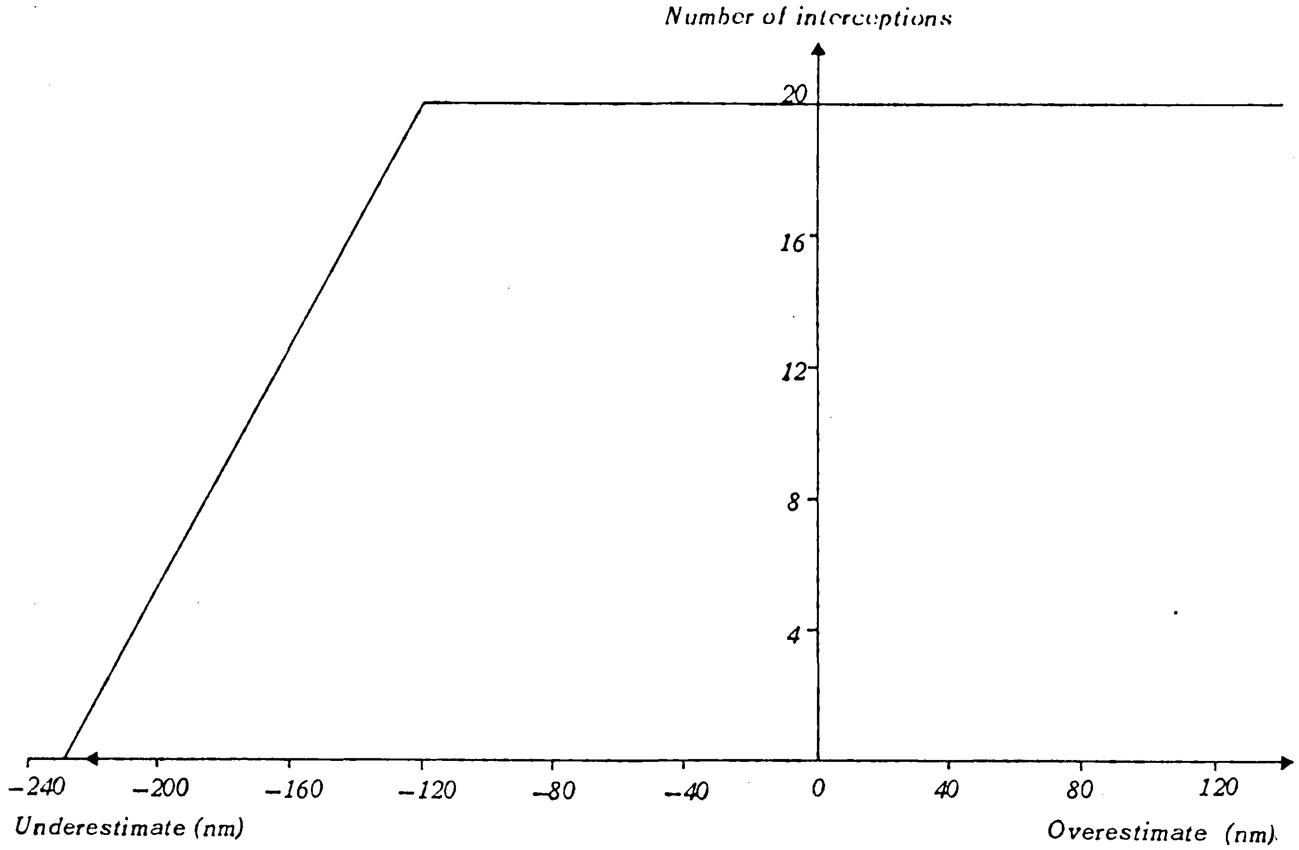


FIGURE B1 SYSTEMATIC ERROR IN THE G.C. ESTIMATE OF THE WARNING DISTANCE OF THE RAID

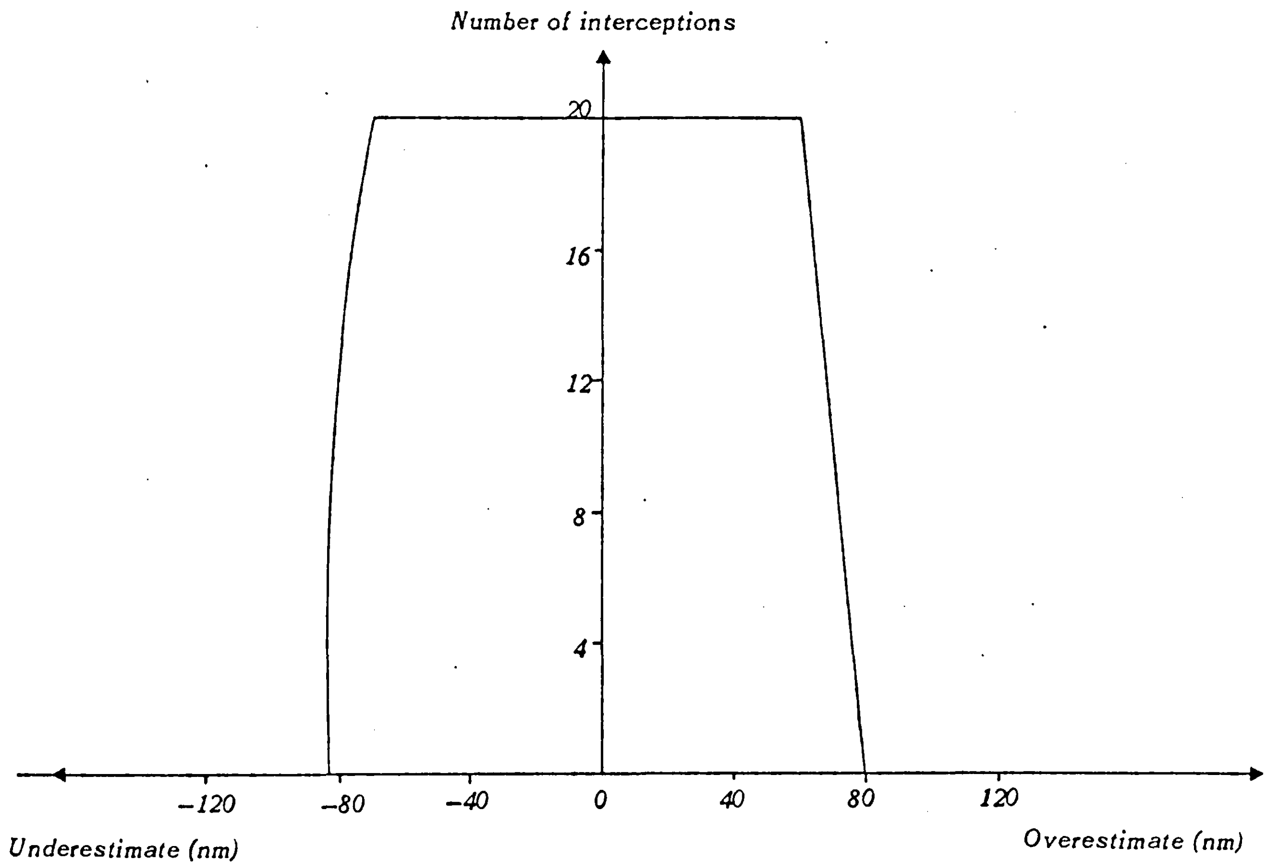


FIGURE B2 SYSTEMATIC ERROR IN THE G.C. ESTIMATE OF THE RAID OFFSET

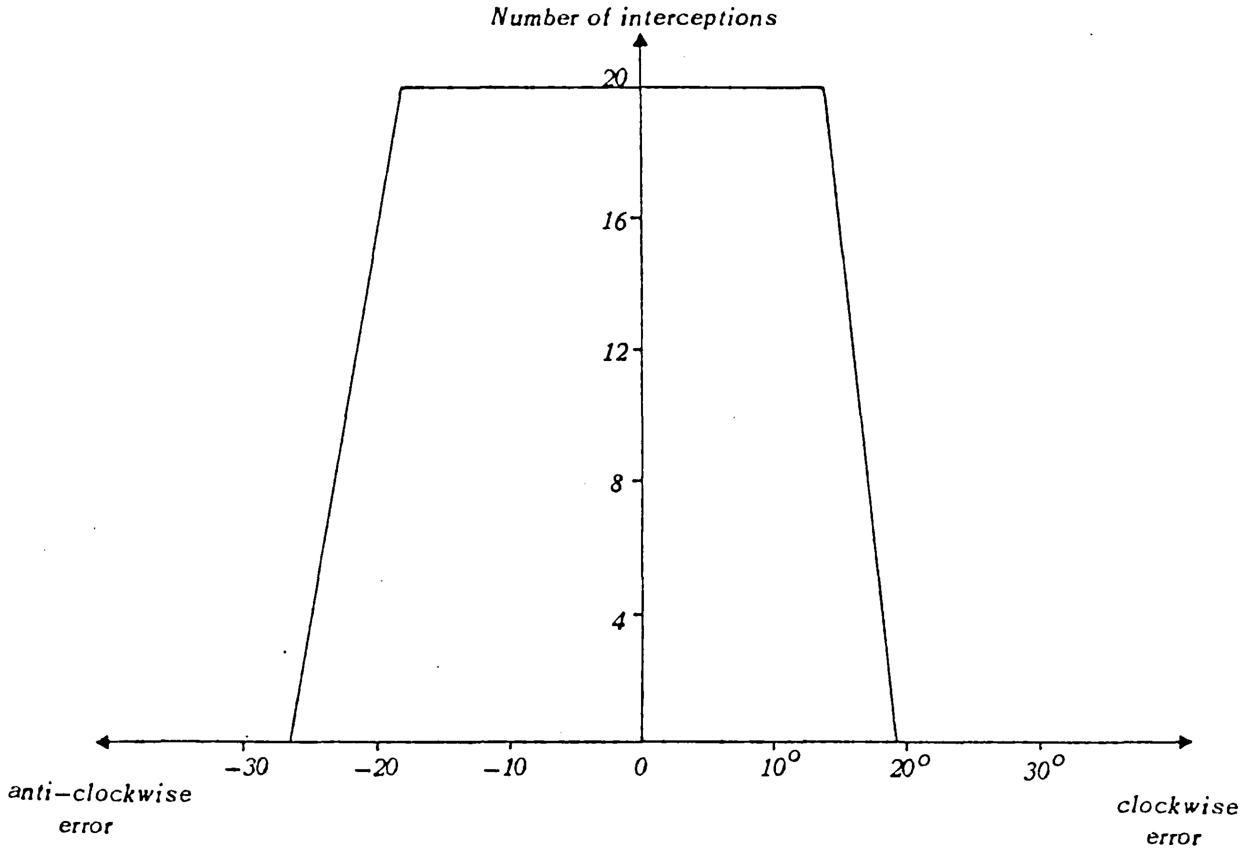


FIGURE B3 SYSTEMATIC ERROR IN THE G.C. ESTIMATE OF THE RAID HEADING

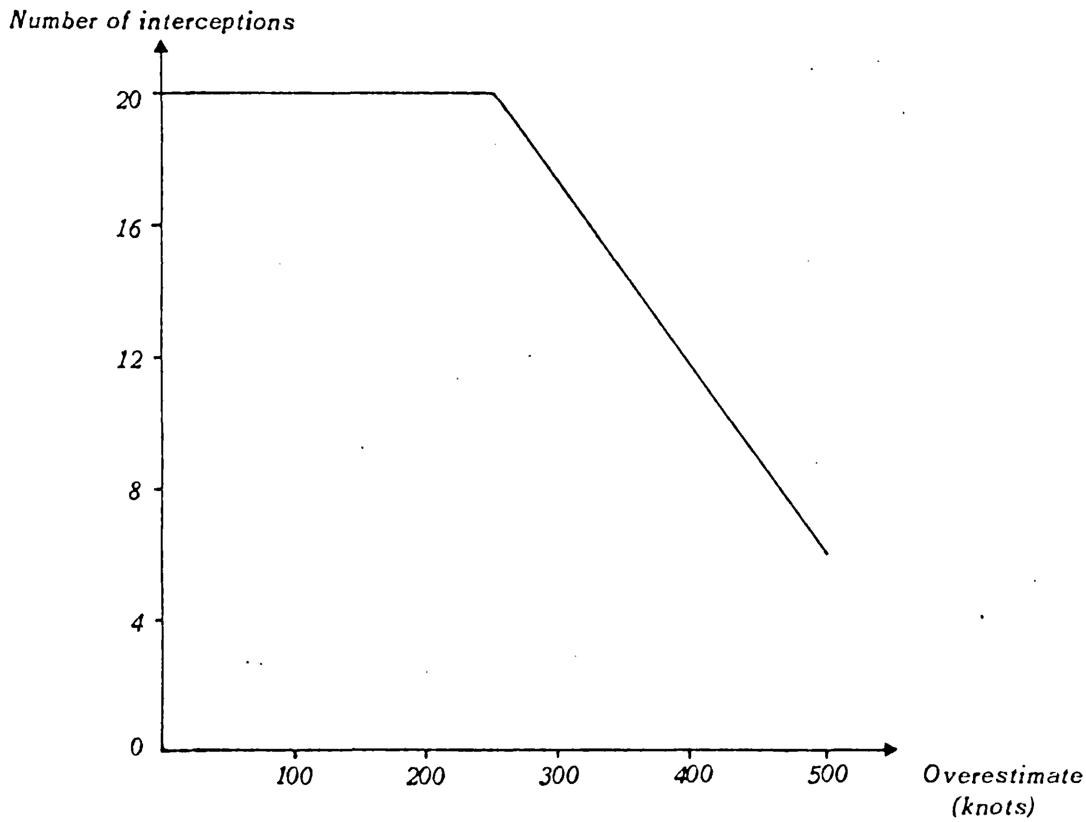


FIGURE B4 SYSTEMATIC ERROR IN THE G.C. ESTIMATE OF THE RAID SPEED

APPENDIX C

FIGHTER DETECTION OF THE RAID

In this Appendix we derive the limiting conditions on fighter detection of the raid. Without loss of generality the coordinate system may be chosen such that the fighter track is along the y-axis, as shown in Figure C1. The fighter has speed V, while the raid travels with speed U and makes an angle ψ ($-\frac{\pi}{2} < \psi < \frac{\pi}{2}$) with the perpendicular to the fighter track. The raid has speed W relative to the fighter, while the raid velocity relative to the fighter makes an angle Φ with the fighter track. This gives

$$W\sin \Phi = U\cos \psi$$

and

$$W\cos \Phi = V - U\sin \psi.$$

Hence

$$W^2 = V^2 + U^2 - 2VU\sin\psi$$

and

$$\tan \Phi = \frac{\cos\psi}{\frac{V}{U} - \sin\psi}$$

The fighter detection area is represented by a sector of a circle, defined by the radar range R and angle of scan $\pm \alpha$. A fighter detects with 100% probability inside this sector while it has zero probability of detection outside it. Five cases then arise in defining the limiting conditions upon fighter detection of the raid. These are denoted by (A)-(E) and are illustrated in Figures C2(a-f). Note that we consider only geometrical constraints on detection and do not examine problems caused, for example, by pulse doppler clutter or track resolution.

The fighter detection capability depends upon the angles α and Φ. Figure C3 illustrates, for given values of α, the ranges of values of Φ in which each of the five cases (A)-(E) hold. This is amplified in Figure C4, which shows the regions in the (α,Φ) - plane ($0 \leq \alpha, \Phi \leq \pi$) in which the five cases apply; it also gives the equations of the boundaries between these regions. The symmetry about the axis $\Phi = \frac{\pi}{2}$ is apparent.

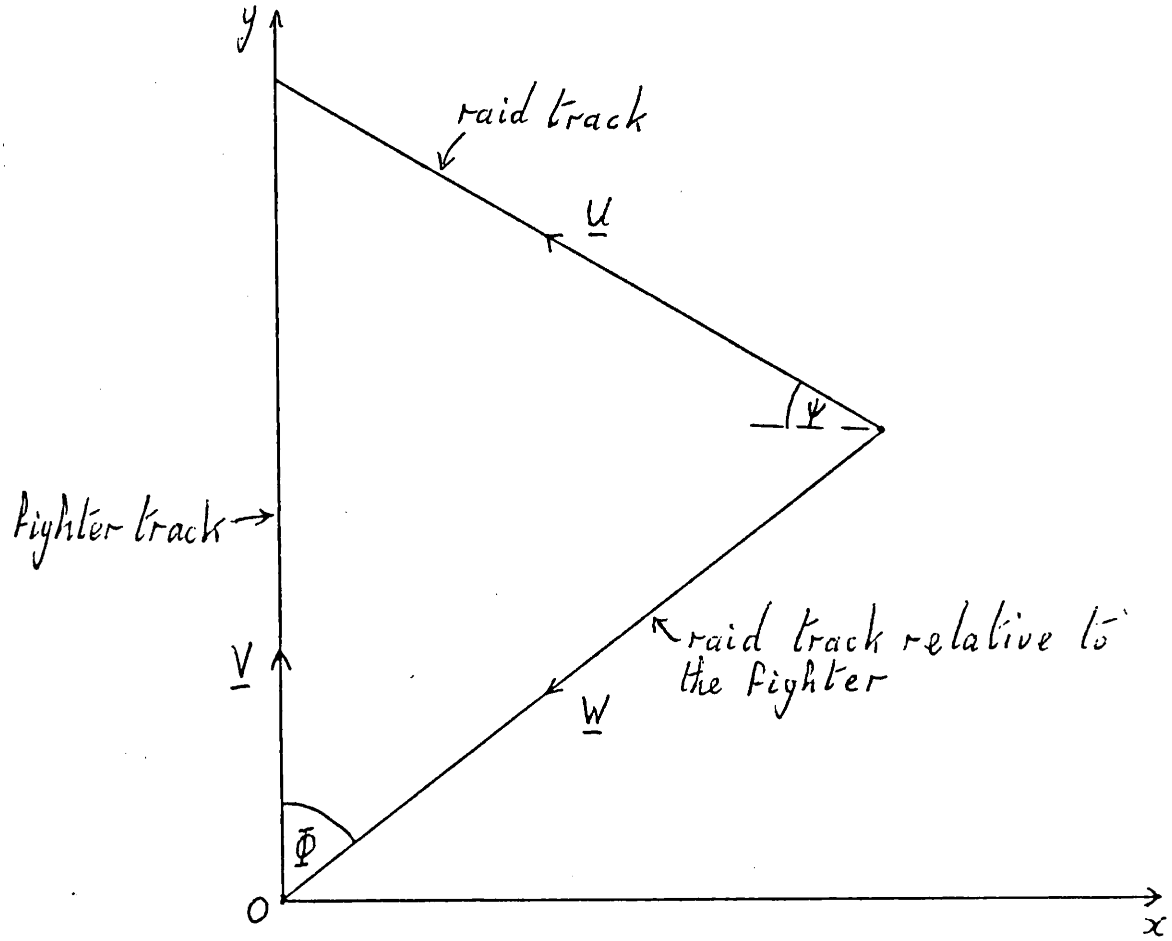


Figure C1. Raid track relative to the fighter

Figure C2(a-f) Limiting conditions on fighter detection of the raid

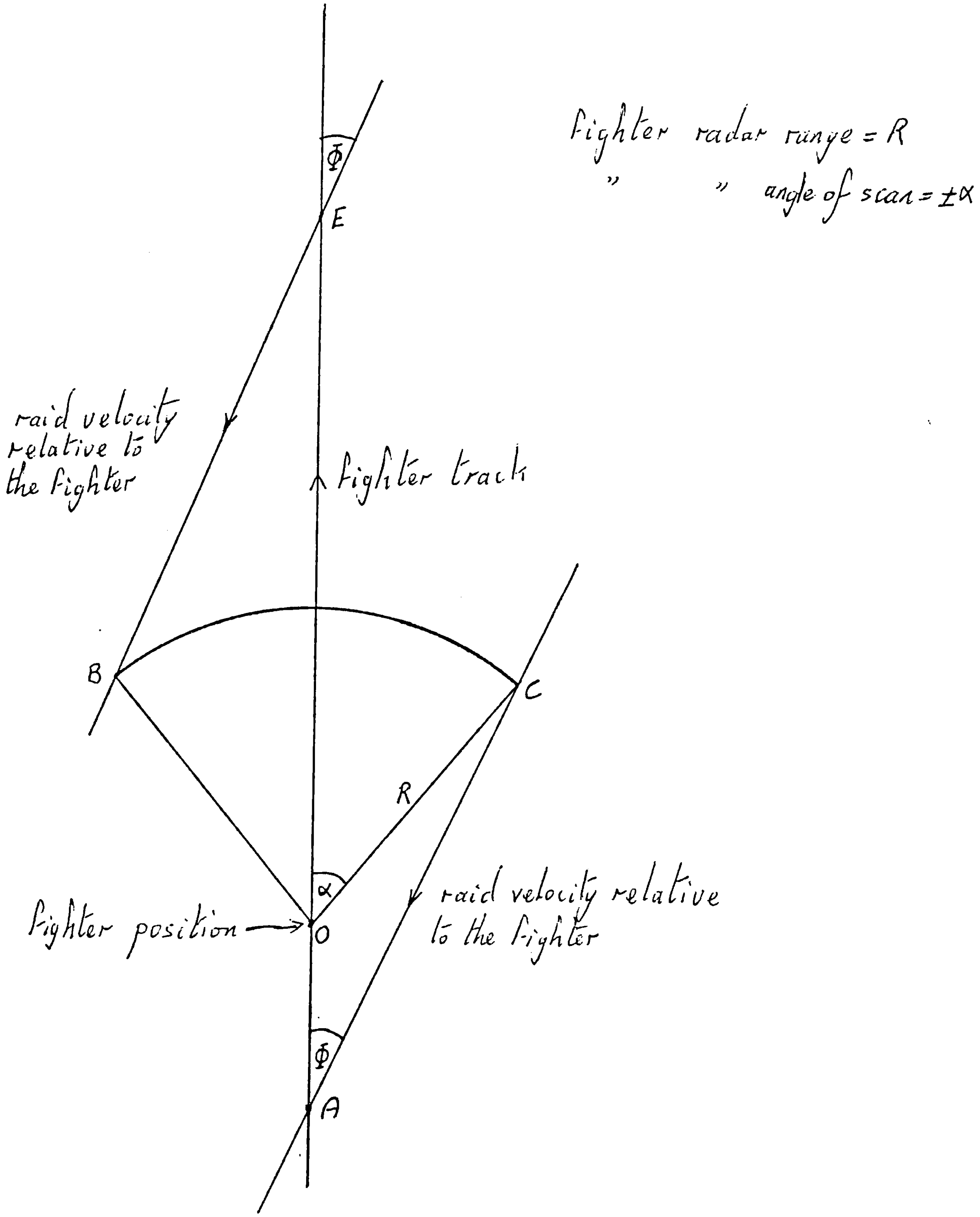


Figure C2(a) $0 \leq \Phi \leq \min(\alpha, \frac{\pi}{2} - \alpha)$; CASE (A)

$\frac{\pi}{2} + \max(\alpha, \frac{\pi}{2} - \alpha) \leq \Phi \leq \pi$

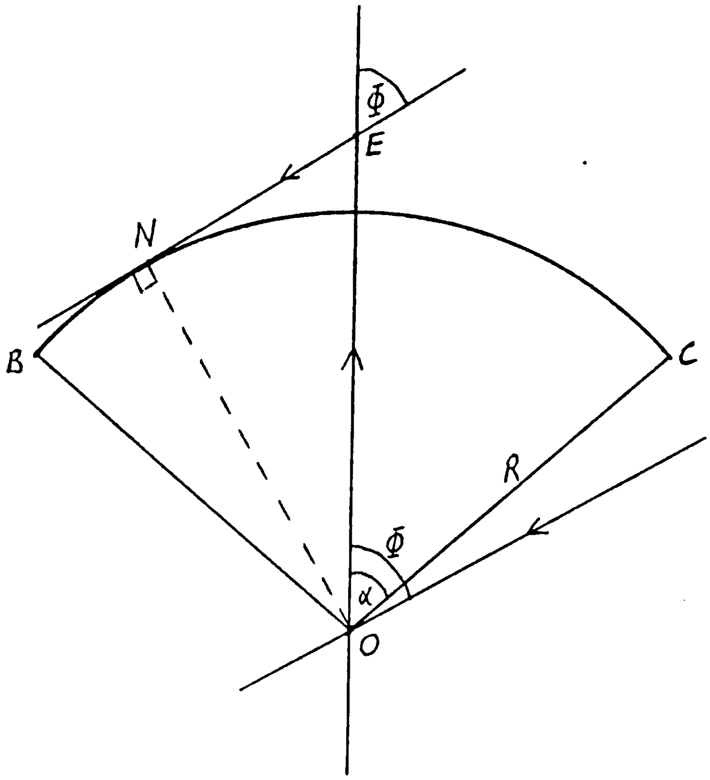


Figure C2 (b) $\max(\alpha, \frac{\pi}{2}-\alpha) \leq \Phi \leq \frac{\pi}{2} + \min(\alpha, \frac{\pi}{2}-\alpha)$ CASE (D)

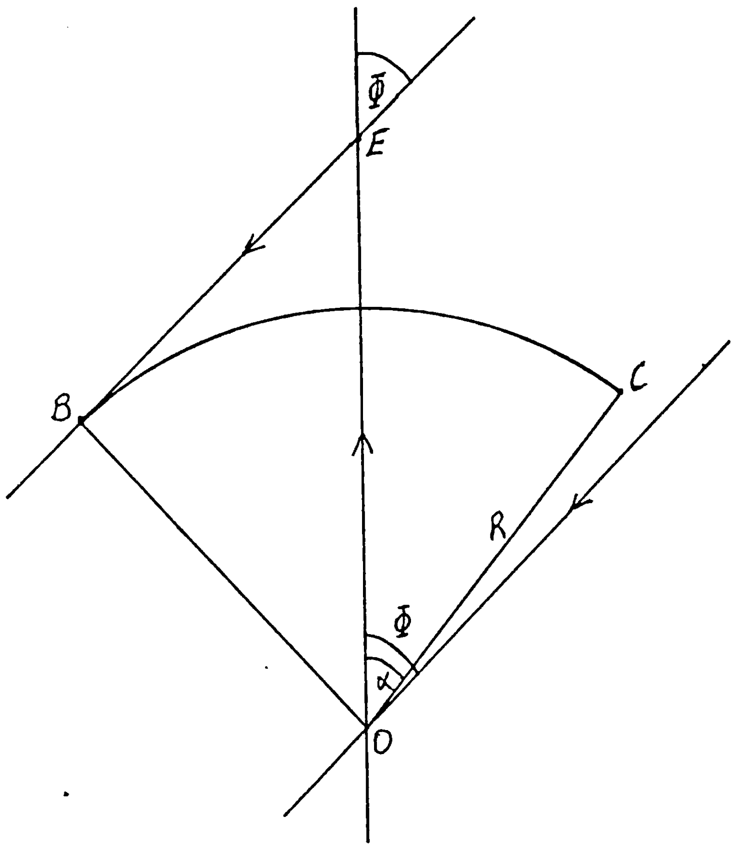


Figure C2 (c) $\alpha \leq \Phi \leq \frac{\pi}{2}-\alpha$ $(0 \leq \alpha \leq \frac{\pi}{4})$ CASE (B)

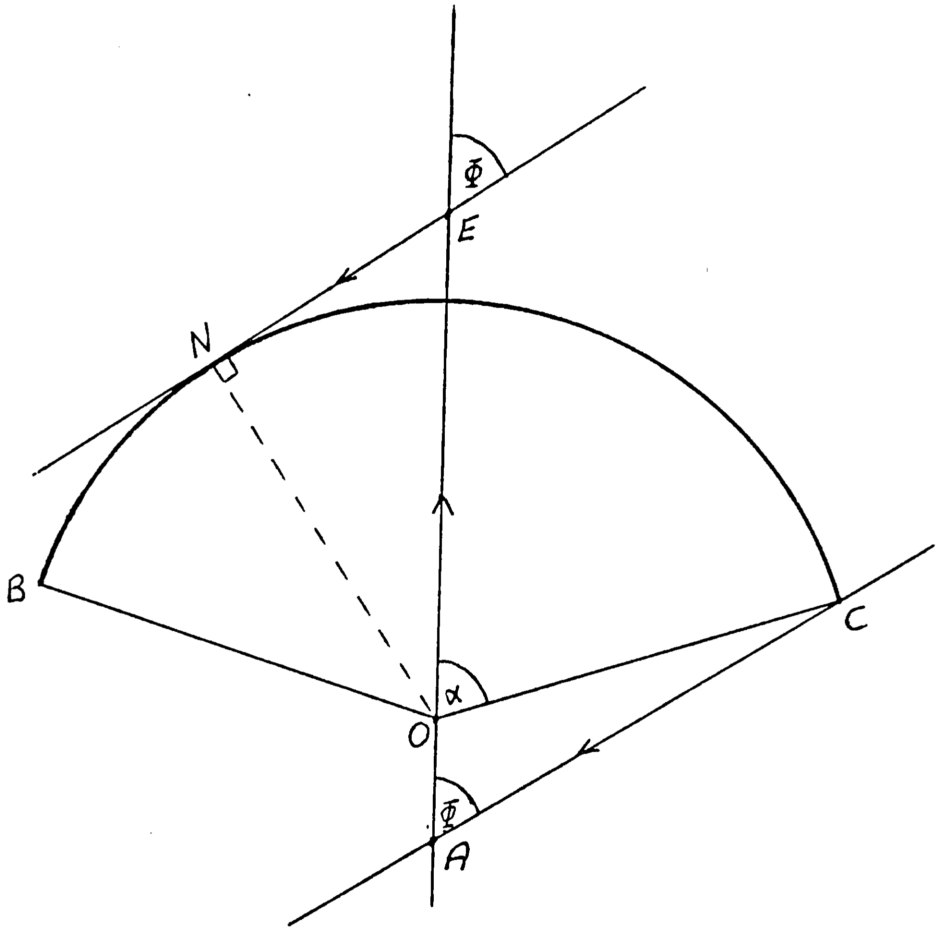


Figure C2(d) $\frac{\pi}{2}-\alpha \leqslant \Phi \leqslant \alpha$ $\left(\frac{\pi}{4} \leq \alpha \leq \frac{\pi}{2}\right)$ CASE (C)

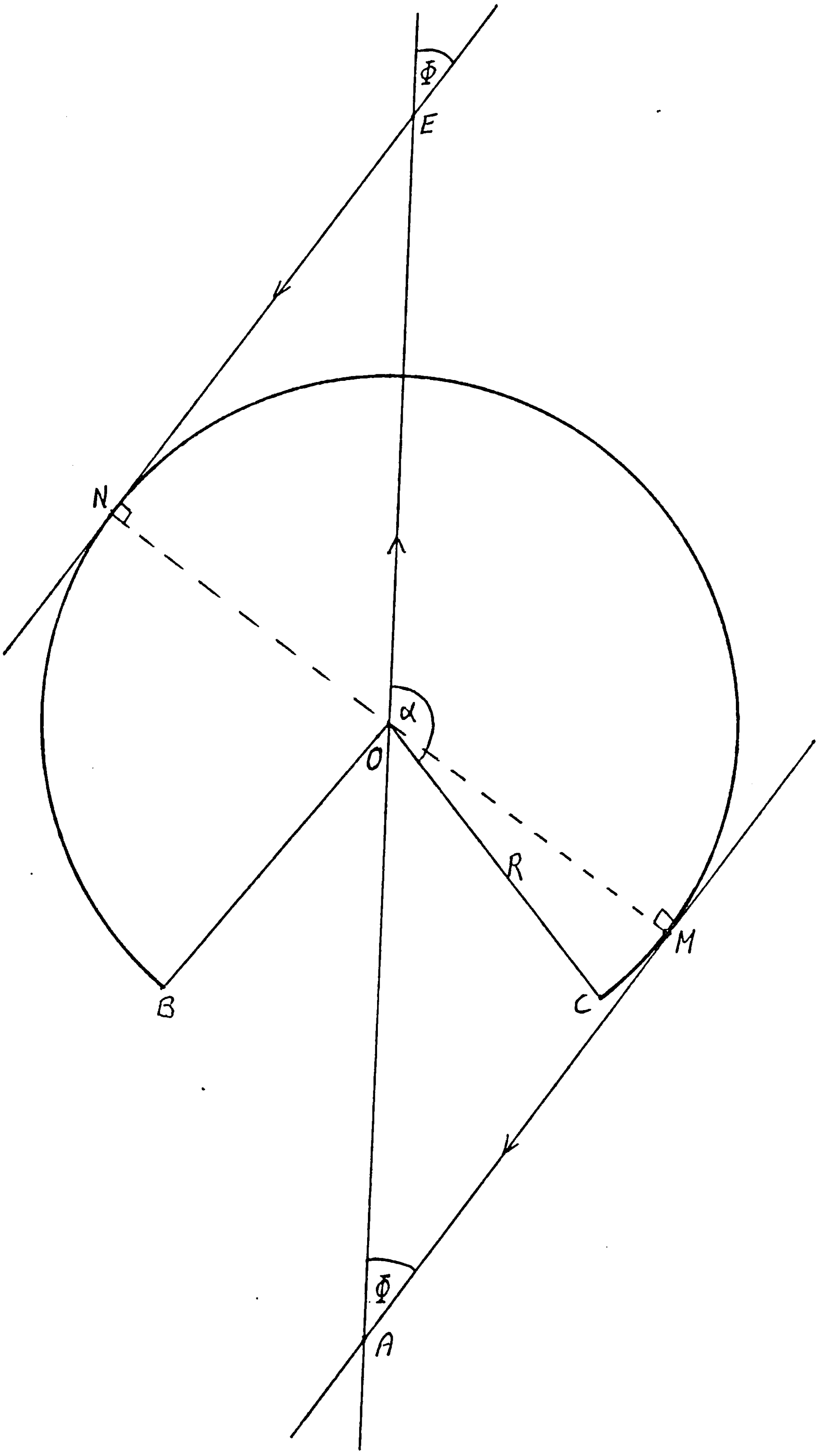


Figure C2(e) $0 \leq \Phi \leq \alpha - \frac{\pi}{2}$; $(\alpha \geq \frac{\pi}{2})$ CASE (E)

$\frac{3\pi}{2} - \alpha \leq \Phi \leq \pi$

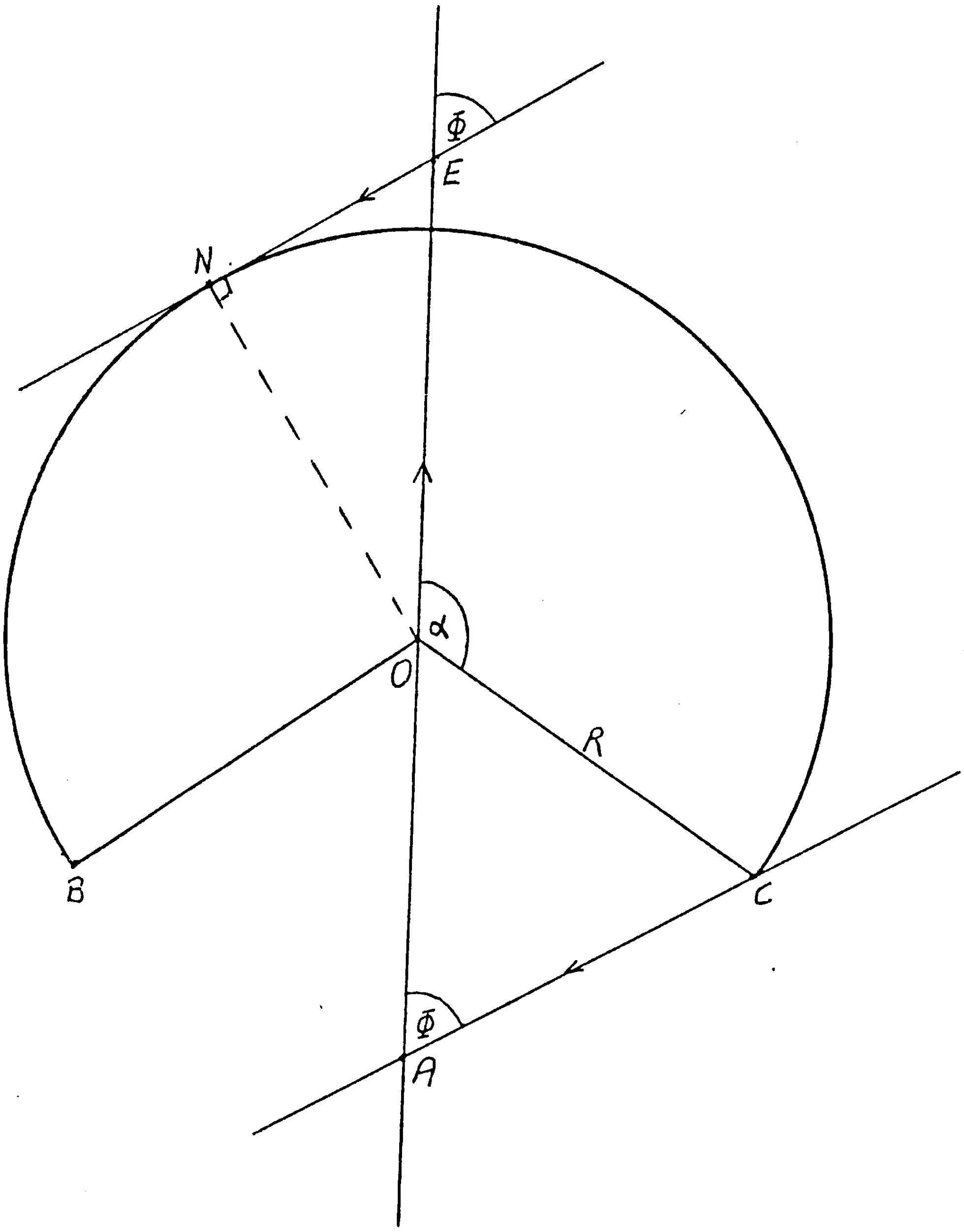
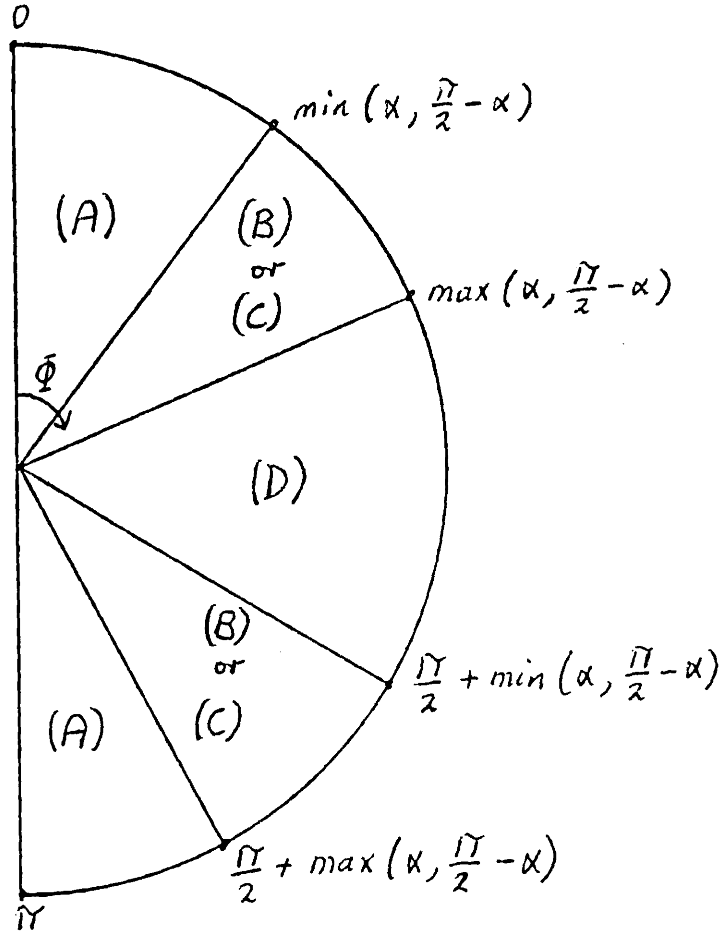


Figure C2 (F) $\alpha - \frac{\pi}{2} \leq \Phi \leq \frac{3\pi}{2} - \alpha$ $\left(\alpha \geq \frac{\pi}{2}\right)$ CASE (C)



Note

(B) holds if $\alpha \leq \frac{\pi}{4}$

(C) " " $\alpha > \frac{\pi}{4}$

Figure C3(a) Ranges of (α, Φ) over which the five cases hold: $0 \leq \alpha \leq \frac{\pi}{2}$

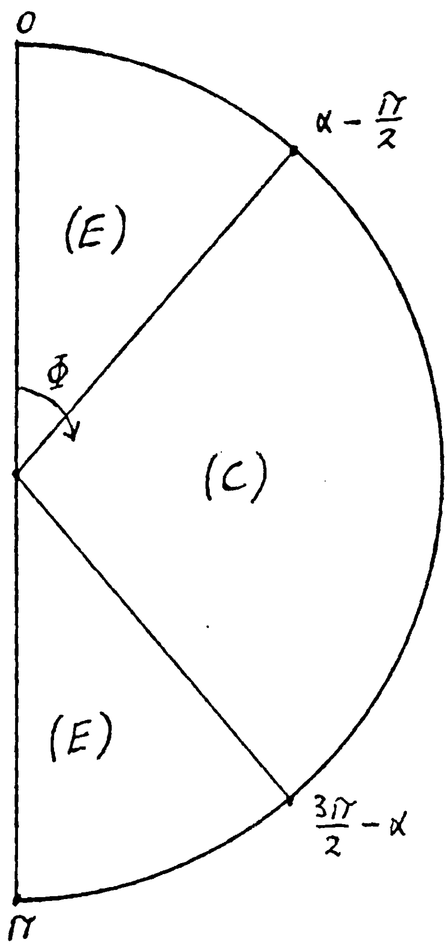


Figure C3(b) Ranges of (α, Φ) over which the five cases hold: $\frac{\pi}{2} \leq \alpha \leq \pi$

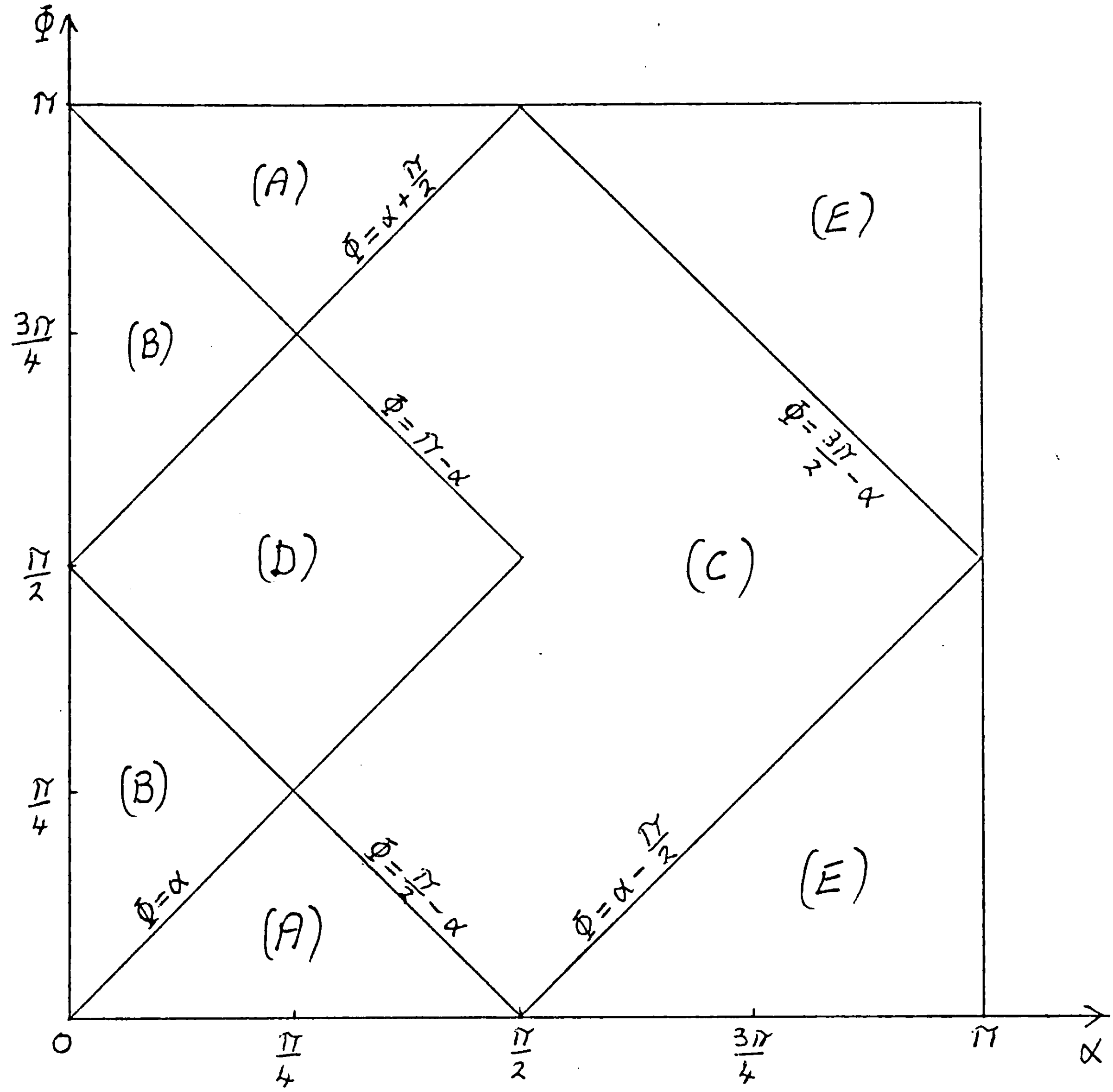


Figure C4. Regions in the (α, Φ) plane in which the five cases hold

REFERENCES

Anderson LB. Brief review of some air-to-air models. Institute for Defence Analysis, Paper WP29, 1972.

Andrews DR. An analysis of the capability of NATO Central Region air defence in a 1982 timeframe. NATO Research Study Group 10. Paper AL243/Panel VII/RSG10/WP2, 1979 (SECRET).

Aris R. Re, k and π : a conversation on some aspects of mathematical modelling. Appl. Math Modelling (1) 386-394, 1977.

Ayker DM and others. The E-3A AWACS: technical performance and integration studies. Shape Technical Centre. TM-470, 1975, (SECRET).

Babich AF, Grason J and Parnas DL. Significant event simulation. Comm. A.C.M. (18) 323-329, 1975.

Beare GC, Daniel DW and Grainger PL. The Battle Group model. DOAE Memorandum 7407, 1974. (RESTRICTED)

Beecham LJ. Theoretical analysis of the kinematics of air combat. Royal Aircraft Establishment TR 72187, 1972. (RESTRICTED)

Berg DJ. Catalogue of war gaming and military simulation models. US Studies Analysis and Gaming Agency, 1980.

Bishop RN and Murray AD. A solution of the radar equation over a curved earth. Admiralty Surface Weapons Establishment WP-XDA-75509, 1975.

Bloomfield JN, Constable PF and Spencer JH. The conventional air threat to the UK Air Defence Region in the late 1980's. DOAE WP 250/1, 1977. (SECRET)

Bloomfield JN, Littlejohn L and Spencer JH. Plans and concepts of operation for the analysis of air defence of the UK in the late 1980's. DOAE WP 250/2, 1977 (SECRET)

Bloomfield JN and Spencer JH. An introductory analysis of the future air defence of the UK, with emphasis on the contribution of sensors to fighter intercept capability. DOAE Report 7707. 1977. (SECRET)

Borgart P — The vulnerability of the manned weapon system. Part I: Probability of detection. <u>International Defence Review</u> (4) 667-671, 1977.

Bok JR — Predicted Sparrow lethality against Soviet fighter aircraft. U.S. Naval Missile Centre, <u>TM-69-58,</u> 1973, (SECRET)

Boulton MF and others — Air Defence study progress report. Phase I - Central Region (2ATAF) Part 1. Shape Technical Centre <u>9980/ORD/S28/72,</u> 1972. (SECRET)

Bourne MJ — The use of interceptors on ground alert and in barrier CAPs to defend shipping. <u>DOAE Note MN 7528,</u> 1975. (SECRET)

Bourne MJ — Consideration of the effect of raid disruption by interceptors. <u>DOAE Note MN 7606, 1976.</u> (SECRET)

Box GEP and Muller ME — A note on the generation of normal deviates. <u>Ann.Math.Statist.</u> (28) 610-611, 1958.

Boyle D and Furlong RDM — NATO AWACS - Now or Never? <u>International Defence Review</u> (1) 43-48, 1977

Boyle D — The problem of identification. <u>International Defence Review</u> (4) 676-680, 1977.

Boyle D — UK Airborne Early Warning. <u>International Defence Review</u> (3) 372-376, 1979.

British Aerospace — Cost Effectiveness of AST 403 aircraft - Part I, Air Combat studies. <u>APN 220 - Part I,</u> 1977. (SECRET).

British Aerospace — A preliminary study of supersonic speed in medium range combat. <u>APN 298,</u> 1980. (CONFIDENTIAL).

British Aircraft Corporation — MRCA ADV Avionics System Definition. <u>Vol. AD-11/2,</u> 1975. (SECRET).

Brown DW and Schemel RE — AWACS deployment in Europe - Aspects of communications vulnerability. Shape Technical Centre, <u>TM-472,</u> 1975. (SECRET).

Burdon D, Soldan P and Domville A — Private communication, EASAMS, 1977.

Burke TJ — The effect of aircraft reliability and maintainability on fighter availability. <u>DOAE WP (Air) 108,</u> 1974. (SECRET).

Central Tactics and Trials Organsiation	ECCM capability of UK air defence system. CTTO/45/20/EW, 1975a. (SECRET).
Central Tactics and Trials Organisation	Phantom AWG-11/12 overland radar detection trial. CTTO/13/12/Air, 1975b. (CONFIDENTIAL).
Central Tactics and Trials Organisation	Phantom Tactical Study - Optimum employment of the Phantom FGR2 in the air defence role in the Central Region. CTTO/13/30/Air, 1976. (SECRET).
Central Tactics and Trials Organisation	The ADGE Tactics Manual. CTTO/6/6/Air, 1978. (SECRET).
Chief Scientist (RAF)	A study of the effectiveness of MRCA and Phantom in the air defence role. Memorandum 388, 1974. (SECRET).
Chief Scientist (RAF)	Interception performance of MRCA and Phantom aircraft against high level targets. Note 7/75, 1975. (SECRET).
Clark BM	Fighter defence of the United Kingdom in 1976, against conventionally armed low level bomber raids. RAF Strike Command, Memorandum 1/75, 1975. (SECRET).
Clark BM, Thomas PH and Saunders NP	Trial 'Flaming Torch'. O.R. Branch, RAF Strike Command, Memorandum 1/78, 1978. (CONFIDENTIAL).
Cochram R	The use of Lanchester's equations to predict the degree of disruption of ground attack missions by fighter patrols. DOAE WP (Air) 47, 1971a. (RESTRICTED).
Cochram R	Sensitivity analysis of factors affecting disruption of ground attack aircraft by fighters. DOAE WP (Air) 51, 1971b. (RESTRICTED).
Cocks C	An aspect of battlefield CAPs. DOAE WP (Air) 7610, 1976. (SECRET).
Collins FI and Guthrie D	A model for the analysis of AEW and CAP aircraft availability. Nev. Res. Log. Quart. (10) 73-79, 1963.
Coucill GC	The modelling of air defence operations: UKADGE. DOAE WP (Air) 124, 1975a. (SECRET).
Coucill GC	The modelling of air defence operations: 2ATAF and ADGE. DOAE WP (Air) 130, 1975b. (SECRET).

Coucill GC and Grainger P — Modelling the Air Defence Ground Environment. DOAE WP (Air) 7612, 1976. (SECRET).

Dare DP and James BAP — The derivation of some parameters for a Corps/Divisional model from a Battle Group model. DOAE Memorandum 7120, 1971. (RESTRICTED).

Dare DP — The NATO Deployment Model, DOAE Memorandum 7210, 1972. (RESTRICTED).

Davies JH and Van Dijk A — NADGE response to the very high altitude high speed threat. Shape Technical Centre TM-483, 1975. (SECRET).

DOAE — Phantom in the maritime air defence role. DOAE WP 110, 1974. (SECRET).

Dockery J, Leiser M and Aitken M — A theatre level simulation with COMO III. Comput. and Ops. Res. (3) 57-71, 1976.

Donovan GS — A survey of battlefield fighter operations. DOAE WP (Air) 7524, 1976a. (SECRET).

Donovan GS — Air inputs to the MCSSG study on the conventional balance. DOAE WP (Air) 7604, 1976b. (SECRET).

Donovan GS and Sands DM — Current state of development of the ADGE model. DOAE WP 590/1, 1977.

Donovan GS and Stead LM — Transportable gound based radars in UK air defence. DOAE Memorandum 78117, 1978. (SECRET).

Douch DA — Description of the Air Campaign model. DOAE Memorandum 7503, 1975. (RESTRICTED).

Driessen HB and Taal J — An investigation into strobe triangulation and tracking for the E-3A AWACS. Shape Technical Centre. TM-488, 1975. (CONFIDENTIAL).

DSTI — Long term air threat assessment for UK air defence. D/DSTI/104/72, 1975a. (SECRET).

DSTI — Soviet technical capability for electronic warfare. Note No.10, 1975b. (SECRET).

Dyer MF — NATO MCSSG study on the conventional balance, UK Report on Phase IV. DOAE Report 7617, 1976. (SECRET).

EASAMS — Air-to-Air Interception Studies. Computer model of Sidewinder AIM-9D. TN41, 1969. (SECRET).

EASAMS Air-to-Air Interception Studies. Sparrow AIM-7E homing against two co-speed targets. TN54, 1971. (SECRET).

EASAMS Airborne Defence Studies. Technical note 5, Fighter Deployment Battle Model - Description and Results, 1973. (CONFIDENTIAL).

EASAMS Maritime Defence Studies. Technical note 1; Capability of CAP and land-based interceptors in a clear air environment, 1976a. (SECRET).

EASAMS Airborne Defence Studies. Interception Model - Program Documentation. Technical note 2322/4/2, 1976b. (SECRET).

EASAMS Airborne Defence Studies. Study of Phantom interception system in an ECM environment. Technical note 2407/3/1, 1977. (SECRET).

Engelbrecht-Wiggans R and Maxwell WL Analysis of the time indexed list procedure for synchronisation of discrete event simulations. Man. Sci. (24) 1417-1427, 1978.

Facey DA and Others The contribution of the E.3A AWACS to air defence in the Central Region. Part I - The Report. Shape Technical Centre. TM-471, 1975. (SECRET).

Fishman GS and Kiviat PJ Digital Computer Simulation: Statistical Considerations. Rand Corporation RM-5387, Santa Monica, California, 1967.

Forder RA Integrated maritime air defence. DOAE Report 7804, 1979. (SECRET).

Forsyth OF and Penick JJ Spears - an AAW performance simulation. Naval Research Labs, Stanford Research Institute. Report 7958, 1976.

George D A description of the computer model to support the Air Campaign Game. DOAE Paper 637/100, 1979. (CONFIDENTIAL).

Gilson C The Tornado F.2 ADV. International Defence Review (6) 865-869, 1978.

Goda HL and Hanks NJ Study of air combat simulation. Institute for Defence Analysis S-381, 1971.

Grainger PL A Proposal for a Model of Air Defence Fighter Operations. DOAE WP (Air) 7509, 1975a. (RESTRICTED).

Grainger PL — Existing Models of Fighter Operations. DOAE WP (Air) 121, 1975b. (RESTRICTED).

Greene TE and Huntzicker JH — Current Techniques at RAND for study of air-to-air combat, RAND Corporation, 1967.

Grenader U — A tactical study of evasive manoeuvres. Research Institute of National Defence, Sweden. FOA Report Vol.2, No.4, 1968.

Gullick AG and others — The contribution of early warning from the FRG low altitude radar system to Central Region air defence. Shape Technical Centre TM-396, 1974. (SECRET).

Gullick AG and others — Study of options for future air defence: North Norway phase. Shape Technical Centre, SN 1979/OR/1, 1979. (SECRET).

Guthrie D and Means EH — Relationships among potential sorties, ground support, and aircraft reliability. Nav. Res. Log. Quart. (15) 491-506, 1968.

Hall CJ — Daily availability of a guided weapon system operating in the CAP role. Rex Thompson and Partners. RTP/600/6, 1974. (CONFIDENTIAL).

Hall JA — An improved pseudo-random number generator. DOAE WP 630/1, 1979.

Happel WJM and Mulders GGM — The COMO - III air battle model: program description. Shape Technical Centre. TM 232, 1970

Harreschou P and Kerr RM — NEWAIR - A theatre-level air model description. Shape Technical Centre. TM 493, 1975.

Hawes KNJ — The influence of electronic warfare on penetration of air defences. DOAE WP (Air) 133, 1975. (RESTRICTED)

Hermann C — Validation problems in games and simulation. Behavioral Sci (12) 216-231, 1967.

Herzog C — The Middle East war 1973. Royal United Services Institute Journal, 3-13, March 1975.

Honningstad TJ and Kerr RM — AIR-2. A theatre level air model description and User's Guide. Shape Technical Centre TM-350, 1973.

Howes DR and Thrall RM — A theory of ideal linear weights for heterogeneous combat forces. Nav. Res. Log. Quart. (4) 645-659, 1973.

James BAP — Aggregating Battle Group level results to provide direct fire attrition coefficients for the Corps Model. DOAE LWP 1151/1, 1976.

James BAP — Concepts for the new DOAE Corps Model. DOAE LWP 584/5, 1977. (RESTRICTED)

Jansson B — Random Number Generators. Victor Pettersons, Stockholm, 1966.

Jordan GHB — The modelling of air defence operations. DOAE WP (Air) 105, 1974. (SECRET)

King R — Initial estimates of aircraft attack effectiveness in the close air support and short term interdiction roles in the mid-1980's. DOAE WP (Air) 7510, 1975a. (SECRET)

King R — A further examination of air-to-ground attack effectiveness with particular reference to the Warsaw Pact. DOAE WP (Air) 7513, 1975b. (SECRET)

Klass PJ — AWACS. Aviation Week and Space Technology, 45-51, 29th April 1974.

Lambeth D — A recommended technique for use in event-driven simulation models. DOAE WP 902/1, 1978.

Lave RE — Timekeeping for simulation. Journal of Industrial Engineering (18) 389-394, 1967.

Law AM — Statistical analysis of the output data from terminating simulations. Nav.Res.Log.Quart. (27) 131-143, 1980.

Law AM and Kelton WD — Confidence intervals for steady-state simulations. I : A survey of fixed sample size procedures; II - A survey of sequential procedures. Dept. of Industrial Engineering, Univ. of Wisconsin. Report Nos. 78-5 and 78-6, July 1978 and March 1979.

Lawrence R and Cairns G — Twin target analogue hybrid simulation results for the Sky Flash missile. Royal Radar Establishment Memorandom 3031, 1976. (SECRET)

Litfin RA — Low Altitude Target Detection. USAF Paper No. 7 1974 CANUKUS Electronics Conference. (SECRET)

Lord WT, Donovan GS and Lee GC — NATO MCSSG Conventional Balance Study (Phase IV): A summary of the U.K. results on the contribution of air support, and comparison with those of the FRG and US. DOAE WP (Air) 7608, 1976. (SECRET)

Lord WT and Spencer JH — Notes on the assessment of the effectiveness of the future air defence of the UK. DDAE Memorandum 77127, 1977. (CONFIDENTIAL)

Lord WT, Bloomfield JM and Spencer JH. — A resume of the current analysis of UK air defence. DOAE WP 615/2, 1978. (CONFIDENTIAL)

Lord WT and Spencer JH. — Exploratory examples of the comparative cost-effectiveness of fighters and shorads. DOAE WP 615/3, 1978. (RESTRICTED)

Lord WT — The background to the form of methodology proposed for Study 264. DOAE WP 615/6, 1978. (RESTRICTED)

Lowell BA and others — IDA Tactical Air Model. Institute for Defence Analysis P1409, 1979.

Macdonald JA — Air defence of the Central Region in 1985. Early warning against low level raids. DOAE WP (Air) 106, 1974. (SECRET)

Mann JR — The Como III Air Battle Model. Shape Technical Centre Professional Paper 9, 1967.

Marsaglia G and Bray TA — A convenient method of generating normal random variables. SIAM Review (6) 260-264, 1964.

Meller R — Europe's new generation of combat aircraft. Part I - the increasing threat. International Defence Review (2) 175-186, 1975.

Midgley PJ — Comparison of MRCA (ADV) and Phantom. Part II : availability/sortie capability. British Aircraft Corporation APN 95, 1974. (SECRET)

Midgley PJ — Assessment of MRCA (ADV) and some competitors in UK, NATO and Naval air defence roles. British Aircraft Corporation APN 140, 1975. (SECRET)

Mihram GA — Some practical aspects of the verification and validation of simulation models. Ops Res Quart (23) 17-29, 1972.

Mihram GA — Simulation methodology. Theory and Decision (7) 67-94, 1976.

Modelski TP — Wasgram: an interactive war game. John Hopkins Univ., Applied Physics Lab., Laurel, Maryland, 1978.

Nance RE — On time flow mechanisms for discrete system simulation. Man Sci: Theory (18) 59-73, 1971.

Nowottny, P — Agreed Fighting Characteristics. Discussion Paper, Scicon Consultancy International Ltd., 1980.

Panyalev G — The SU-19 Fencer Threat to Western Europe. International Defence Review, 67-69, 1976.

Parkinson VL, Thomas PH and Clark BM — Analysis of the air defence aspects of Exercise Highwood. Strike Command, ORB Memorandum 7/78, 1978. (CONFIDENTIAL)

Rasmussen PH — The Soviet Naval Air Force - Development, Organisation and Capabilities. International Defence Review, 689-695, 1978.

Reed WE — ECM and ECCM. A bibliography with abstracts, 1964-1974. US National Technical Information Service PS-75/165, 1974.

Richardson HD — The Automation of the Battlegroup - Corps Model Interface. DOAE LWP 7610 1976. (RESTRICTED)

Richardson HD — Assumptions in the new DOAE Corps Model. DOAE WP 584/13, 1977. (RESTRICTED)

Richardson HD and Dunkerley PL — Concepts for the Representation of Air Support in the new DOAE Corps Model. DOAE WP 584/12, 1977. (RESTRICTED).

Roros JK — NEWAIR Test Game. Shape Technical Centre, TM-520, 1976. (SECRET)

Rose R and Hogsflesh RA — Comparative performance for the MRCA/AD and Phantom at low altitude. Royal Aircraft Establishment TM 1576, 1974. (SECRET)

Royal Aircraft Establishment — The detection and interception capabilities of the Phanton and MRCA air defence aircraft. Memorandum WE 1397, 1974a. (SECRET)

Royal Aircraft Establishment — The performance of XJ 521 in an ECM environment. Memorandum WE 1401, 1974b. (SECRET)

Schenk W — The AIM-9L Super Sidewinder. International Defence Review, 365-369, 1976.

Seyb EK — A mathematical model for the calculation of visual detection range. Shape Technical Centre TM-152, 1967.

Shephard RW — War Gaming as a technique in the study of Operational Research problems. Opl. Res. Quart. (14) 119-130, 1963.

Shipman KEW — Airborne Defence Studies. Fighter Deployment Battle Model - threat evaluation and fighter allocation. EASAMS, WP No. 3, 1974. (RESTRICTED)

Skolnik MI — Introduction to Radar Systems. McGraw Hill, 1962.

Smith J — Computer Simulation Models. Griffin, 1968.

Sparre E — Fighter Operations in the Low Altitude Defence Role: The Detection Problem. Shape Technical Centre TR 84, Vols I-III, 1971-72. (SECRET)

Sundaram GS; Loomis R; Eustace HF — International Defence Review, February 1976.

Sutcliffe PM — MRCA in the maritime air defence role. DOAE WP 109, 1974. (SECRET)

Sutherland JW — New data handling technology in air defence. International Defence Review (4) 616-618, 1976.

Thornton RC — The Ground Based Air Defence Model - EVA: Physical Representation. DOAE WP 579/1, 1977. (RESTRICTED)

Thornton RC — The Ground Based Air Defence Model - EVA: Data Requirements and Outputs. DOAE Memorandum 77109, 1978. (RESTRICTED)

Tocher KD — The Art of Simulation. English Universities Press, 1963.

Ulvila JW and Brown RV — Step-Through Simulation. OMEGA (6) 25-31, 1978.

Vaucher JG and Duval P — A Comparison of Simulation Event List Algorithms. Comm. ACM (4) 223-230, 1975.

Vaughan WS and Virnelson TR — The Development of an Information System for Identification and Raid Recognition in Air/Space Defence. Volume I : Initial Steps. U.S. Govt. Research Report AD-600 538, 1963.

Vincent TL, Sticht DJ and Peng WY — Aircraft Missile Avoidance. Ops Res. (24) 420-437, 1976.

Watts JW — Optimum Deployment of Combat Air Patrol and Airborne Early Warning aircraft in defence of a suface task force. Directorate of Naval Operational Studies, Note 6/76, 1976. (RESTRICTED)

Welp DW and Brown RA — Plan for Development of the Air-to-Air Avionics Evaluation Program. Battelle Columbus Laboratories, Columbus, Ohio, 1974.

Welp DW, Brown RA and Rea FG — Development of an Avionics Evaluation Program for Air-to-Air Mission Analysis. Air Force Systems Command AFAL-TR-75-159, 1975 (RESTRICTED)

Whitaker, AW — Effect of Air Combat on Blue Close Support Operations. DOAE WP (Air) 42, 1970. (SECRET)

Witts AD — The Corps Model. DOAE Memorandum 7408, 1974. (RESTRICTED).

Wong PJ — Reachable Sets for Tracking. Ops Res (22) 497-509, 1974.

Wyman FP — Improved Event Scanning Mechanisms for Discrete Event Simulation. Comm ACM (18) 350-353, 1975.