AD-A255 628

(1)

THE HUNTER-KILLER MODEL

VERSION 2.0

USER'S MANUAL

Ross Fairbrother
Robert T. Campbell

Science Applications International Corporation
Computer Modeling and Simulation Division
1710 Goodridge Drive
McLean, Virginia 22102

1 December 1986

Contract No. DAAK21-85-C-0058

Prepared for

The Center for Night Vision and Electro-Optics
DELNV-V
Fort Belvoir, Virginia 22060

SAIC
Science Applications International Corporation

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) The Hunter-Killer Model User's Manual	5. TYPE OF REPORT & PERIOD COVERED Final
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Ross Fairbrother, Robert T. Campbell	8. CONTRACT OR GRANT NUMBER(s) DAAK21-85-C-0058
9. PERFORMING ORGANIZATION NAME AND ADDRESS Science Applications International Corporation, 1710 Goodridge Drive, McLean, VA 22102	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS The Center for Night Vision and Electro-Optics Laboratory DELNV-V, Fort Belvoir, VA 22060	12. REPORT DATE December 1, 1986
	13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

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

18. SUPPLEMENTARY NOTES

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

Hunter-Killer Model

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

This manual describes the methodology of the Hunter-Killer Model. Each algorithm used by the model is explained and the source of any equations is provided. The manual provides directions for constructing a data set. The manual also provides procedures to run the model. The reports generated by the model are explained.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

FIGURES

1.0 INTRODUCTION

1.1 PURPOSE

The Hunter-Killer Model Version 2.0 User's Manual is designed to provide a detailed explanation of the concepts involved in the model's design. Additionally, it includes directions for preparing the data required by the model and for executing the model. This manual should contain all of the information required by most users of the model. In addition to this manual, documentation of the model includes:

The Hunter-Killer Model Version 2.0 Executive Summary. This manual provides managers with an overview of the features and capabilities of the Hunter-Killer Model. Data and computer requirements are described.

The Hunter-Killer Model Version 2.0 Programmer's Manual. This manual provides a detailed explanation of the internal structure of the model. It includes a dictionary of all global variables occurring in the model, a description of each program, the coding standards used, and instructions for modifying and compiling the model. This manual will be of interest primarily to the model's maintenance programmer.

A database system has been developed for the Hunter-Killer Model. This system is described in Section 3.2. The documentation for the database system consists of:

The Hunter-Killer Model Version 2.0 Database System User's Manual. This manual describes how to use the database system to store data and to prepare run streams for the model.

The Hunter-Killer Model Version 2.0 Database System Programmer's Manual. This manual describes the implementation of the database system. It is intended to be used by a maintenance programmer.

1.2 SECURITY AND CLASSIFICATION

The Hunter-Killer Model source code is UNCLASSIFIED, as are all of the manuals mentioned above. The data used by the model are also, as a rule, UNCLASSIFIED; however, inclusion in a run of the model of sensors or weapons whose performance characteristics are classified would cause the data and output for that particular run to become classified.

1.3 INQUIRIES

Inquiries concerning the Hunter-Killer Model or the Hunter-Killer Database System should be addressed to:

The Night Vision and Electro-Optics Center
DELNV-V
Fort Belvoir, VA 22060

(703) 664-5845

2.0 MODEL METHODOLOGY

2.1 OVERVIEW

The Hunter-Killer Model is designed and constructed to study the performance of electro-optic sensor systems in a combat scenario. The model simulates a two-sided battle between individual platforms. Each platform may have one weapon and up to two sensors, a hunter and a killer. The killer sensor is used to acquire targets and to operate the weapon. In addition, there may be a second, independent sensor called the hunter that supports the killer by acquiring targets and passing them to the killer. This pair of sensors, the hunter and the killer, cooperatively search for targets with start-of-the-art and future sensor systems. Acquired targets are processed through a target selection scheme and are either discarded or engaged by the killer. The targets themselves are hunter-killer platforms, therefore making the model fully two-sided.

The Hunter-Killer Model is capable of simulating a wide variety of military scenarios. The model is open ended in terms of the number of platforms that can be included. The only absolute bound is the available computer memory and time. From a practical point of view, twenty to thirty platforms for twenty replications seems to be an upper bound.

The Hunter-Killer Model may be employed as a tool to perform many types of sensor performance analyses. Some analyses appropriate foı the Hunter-Killer Model are:

Evaluate sensor effectiveness as measured by acquisition time.

Evaluate the sensor contribution to combat as measured by engagement times.

Evaluate the sensor/laser rangefinder contribution to combat as measured by shots fired per kill.

Evaluate sensor performance in terms of combat results such as equipment kills and force ratios.

Evaluate such concepts as Battlefield Identification Friend or Foe (BIFF).

Evaluate variations in search strategy.

The Hunter-Killer Model is basically a collection of discrete events and processes of different types supported by the relationships of modeled military activities. The data describe the equipment, weapons, sensors, and the parameters that depict their performance and relationships to each other. In this manner the combat process between two opponent forces is simulated.

The Hunter-Killer Model methodology is discussed in detail in the following sections. The first sections discuss the components of the model -- the battlefield, platforms, sensors, and so on. These sections are:

Terrain
Platforms
Movement
Air Tactics
Weapons
Laser Rangefinders
Sensors
Johnson Criteria
Acquisition Times and Levels
Automatic Target Recognition Systems

The remaining section, The Simulation, describes how these components are combined to simulate target acquisition and combat.

Due to the manner in which this manual was printed, exponentiation is indicated through computer notation instead of superscripts; i.e., a ** b represents the value a raised to the power b. Special symbols such as Greek letters and the square root sign are not used.

2.2 TERRAIN MODELING AND LINE OF SIGHT

2.2.1 Line of Sight

Line of sight (LOS) between two platforms means that there is a direct and unobstructed visual line between them. The Hunter-Killer Model's terrain module is used to determine the existence of LOS.

LOS is always two-way; that is, if one platform has LOS to another, the second platform also has LOS to the first. The two platforms may not, however, be able to see each other equally well. If the first platform is in defilade and the second is in the open, 100% of the second may be visible to the first while only a fraction of the first is visible to the second.

The existence of LOS between two platforms does not imply that the platforms are able to acquire each other. A platform may have LOS to a target but not be able to make an acquisition if the target is out of the sensor's range or if the target is not in the platform's field of search.

2.2.2 The Battlefield

The Hunter-Killer Model's battlefield is a rectangular coordinate system where coordinates are measured in meters. Only the first quadrant, with positive X and Y axes, is used. Movement vectors and search azimuths are given by compass directions, where the positive Y-axis direction is north (0 degrees), the positive X-axis direction is east (90 degrees), and so on. Superimposed on this battlefield is the terrain model which is used to determine whether line of sight exists between any two hunter-killers.

The platforms are arrayed on the battlefield by giving their X,Y coordinates. Movement for a platform is determined through a set of movement vectors, each of which gives a direction and a distance to be moved along with the platform's speed. After a platform has moved the distance given by a vector, it switches to the next vector in its movement set, if there is one.

The Hunter-Killer Model has available two terrain modeling techniques, which are referred to as low and high resolution terrain. Either technique may be used to determine whether line of sight exists between any two platforms. The low resolution terrain is modeled with a stochastic technique using input mean times that line of sight between platforms is established or broken. The high resolution terrain is modeled through elevations given at fixed intervals. The resolution used is determined through the data input to the model.

Both the low and high resolution techniques model macro-terrain -- the determination of LOS over large distances based on large terrain features. The Hunter-Killer Model additionally allows the modeling of micro-terrain, also called defilade: this is the use by a platform of terrain features to mask itself from others. This micro-terrain is modeled through a curve which gives the fraction of a target that is visible as a function of range. Use of this curve is controlled through an input switch which determines if targets are in defilade.

2.2.3 Low Resolution Macro-Terrain

Low resolution terrain is stochastically simulated through the use of an initial line of sight curve plus two time values. These two values are the mean times that line of sight exists or is broken.

The initial line of sight curve gives the probability of LOS existing between two ground platforms as a function of range. The curve is defined through a set of points, each of which gives a range and the probability of LOS at that range. The number of points used to define the curve is determined by the user. Upon initialization of a replication of the model, the distance between each pair of ground platforms is computed. The probability of LOS is interpolated from the curve. Then, the SIMSCRIPT pseudorandom number generator is used to obtain a sample with a value between zero and one. If the interpolated probability is greater than this value, LOS is said to exist between the pair of platforms. Otherwise, LOS does not exist.

Once the initial LOS state for a pair of platforms is determined, the duration of the condition is computed through a sample made from the exponential distribution using the input mean time LOS exists or is broken, as appropriate. At the end of this interval the LOS state between the pair of platforms is switched to its opposite. A new duration is computed. This process repeats continuously until one of the platforms is killed.

When it is necessary to determine if LOS is possible between two platforms, the current LOS state for that pair of platforms is used. If LOS exists, 100% of the target is visible unless that percentage is later reduced due to micro-terrain.

Note that the simulation is not affected by each change of the LOS state between a pair of platforms changed. Only the state at the time of a move, when all search sectors and target acquisition times are updated, is relevant.

The above algorithm is constantly changing the LOS state between a pair of platforms independent of movement by either of them; however, actions that require the existence of LOS, such as acquiring or engaging a target, are not affected. When a platform moves the LOS state is checked and, if LOS exists to a target, an acquisition is scheduled. The above algorithm may cause LOS to be broken and re-established several times before the acquisition actually occurs, but that does not affect the acquisition. In particular, the LOS state is not rechecked when the acquisition occurs. Similarly, if after a move LOS does not exist between a pair of platforms, no target acquisitions are scheduled. If the LOS state changes before the next move so that LOS exists, the model still does not schedule target acquisitions.

The above description of low resolution terrain only applies to pairs of ground platforms. Line of sight for aerial platforms such as helicopters is handled differently. It is assumed that a platform of this class flies at a low altitude so as to be masked by terrain features while it is maneuvering to a target. It then unmasks to acquire and engage targets, then remasks. When masked, the platform is unable to detect any other platform and cannot be detected itself. When unmasked, the platform has LOS to every ground platform and to every unmasked air platform that is in its search sectors. Similarly, all ground platforms and unmasked air platforms which have the aerial platform in their search sectors have LOS. The ground platforms are 100% visible unless the percentage visible is reduced due to micro-terrain. An unmasked air platform is always 100% visible.

When it is necessary to determine if LOS is possible between a ground platform and an aerial platform, the current masked or unmasked state of the aerial platform is determined. If it is masked, LOS is not possible. LOS between two aerial platforms requires that both be unmasked.

The masking and unmasking of aerial platforms is discussed in Section 2.5.3.

2.2.4 High Resolution Macro-Terrain

The high resolution parametric macro-terrain is modeled using digitized terrain. Digitized terrain consists of terrain descriptions, such as elevations, given at fixed intervals across a map. Digitized terrain maps are available from the Defense Mapping Agency (DMA) in the form of computer tapes. The Level 1 tapes, which are easily obtainable for most of the world, give terrain elevations grouped in squares of one degree of arc on a side. Elevations within these squares are given every 3 seconds of arc along a line of longitude, or approximately 92.6 meters. The elevations along a line of latitude are given every 3, 6, 9, 12, or 18 seconds of arc, depending on the latitide. These intervals are referred to as the LOS intervals. The values are input to the model. The Programmer's Manual contains directions for obtaining and reading a DMA Level 1 Terrain tape.

The data are used to superimpose a grid consisting of rectangles on the Hunter-Killer battlefield. The purpose of the elevation data is solely to provide relatively realistic LOS information. The terrain is ignored for all other purposes, such as movement. Consequently, for the purposes of this model the curvature of the earth may be ignored and the spherical grid may be treated as if it were planar. Doing so introduces a small error in elevation computations of less than one meter at a three kilometer range. (The standard battlefield is approximately 4000 by 2000 meters at its maximum, with engagements occurring at ranges of less than 2000 meters.) With the available relatively coarse data, this is considered to be within the resolution of the model. If finer data become available, the model can be modified to take into account spherical effects.

Each lattice point on the grid has an associated elevation. For all other points on the battlefield, linear interpolation is used to compute the elevation. Figure 2-1 shows a single grid rectangle. Elevations are input for points (X1,Y1), (X2,Y1), (X1,Y2), and (X2,Y2). To determine the elevation at point (X,Y), the elevation at (X,Y1) is computed by interpolating between the values at (X1,Y1) and (X2,Y1). Next, the elevation at (X,Y2) is computed by interpolating between (X1,Y2) and (X2,Y2). Finally, a third interpolation between (X,Y1) and (X,Y2) is performed to compute the elevation at (X,Y).

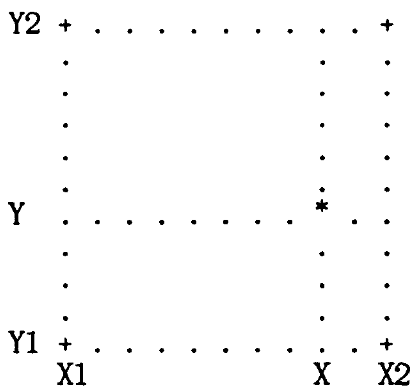

Figure 2-1, Interpolation of an Elevation

For a given pair of platforms, an observer-target (O-T) line is drawn between them, as shown in Figure 2-2. The terrain elevations are computed at the observer, at the target, and at each of the points a through g where the O-T line crosses a grid line.

Figure 2-3 shows a vertical cross-section of the terrain along the O-T line. The elevation angles necessary for the observer to see the target's top and bottom are determined. For any one of the points a through g, LOS exists if the elevation at that point is less than the elevation of the observer's line of sight to the target's top at that point. If the elevation of the point is less than that of the LOS to the target's bottom, the entire target is visible. An elevation greater than that of the LOS to the target's bottom but less than that of the LOS to the target's top blocks a portion of the target. In this case, the percentage visible is calculated. LOS must exist at all points for the observer to see the target. As with low resolution terrain, this computed value may then be reduced if micro-terrain is modeled.

The calculations are essentially identical for ground and air platforms. The only difference is that for an air platform, its altitude as obtained from its movement vector is added to the terrain elevation at its location to determine platform's elevation.

If a platform moves off of the area for which elevation data exist, the model does not abort with an error. Instead, a message is written indicating the platform is not on the data grid. The map is then reflected about its boundaries. For example, assume the map runs from 0 to XN and from 0 to YN. The elevations for points along X = -1 are the same as those along X = +1. Similarly, the elevations along X = -XN are the same as those along X = +XN. The elevations along X = XN + 1 are identical to those at X = XN - 1. If a platform moves even further off the grid, the map is again reflected about its boundaries so that the elevations along X = 2 XN + 1 are the same as those along X = 2 XN - 1, which are the same as for X = 1. Similar rules are used for points off the data grid in the Y-direction. This scheme allows the model to continue when a platform moves off the grid and also avoids any gross discontinuities between altitudes on either side of a grid boundary.

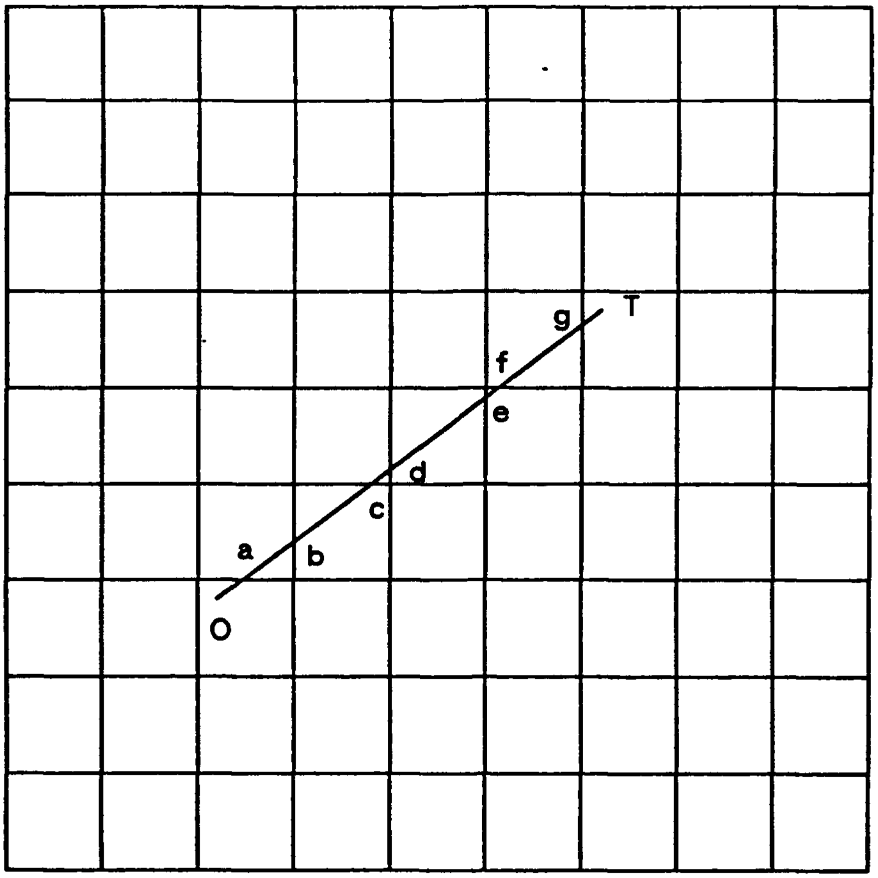

Figure 2-2, Elevations Data Grid

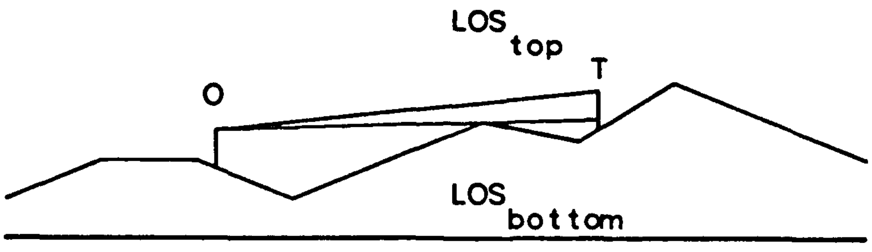

Figure 2-3, Line of Sight Cross-section

2.2.5 Micro-Terrain

After either the low or high resolution macro-terrain module determines that LOS is possible between an observer and a ground target, the fraction of that target that is visible may be reduced through the use of defilade. The modeling of the use of small terrain features to produce as much masking as possible is referred to as micro-terrain or defilade.

A control in the data set is used to determine if defilade is used. If so, curves are input that define the micro-terrain. Curves are defined for each observer class -- ground or air -- and type target combination. Each curve is defined through a set of points, where each point gives a range and the fraction of the target visible at that range. The number of points used to define each curve is determined by the user. If micro-terrain is used and if a curve is not given for a particular observer class and type target pair, it is assumed that the target is able to mask itself from the observer at all times. This means that the fraction of the target visible to that class of observers is always zero.

When micro-terrain is used, the observer class and target type are determined and the correct defilade curve is selected. The distance between the observer and the target is computed. The percentage of the target visible is then interpolated using the selected curve.

The overall percentage of the target that is visible to the observer is the minimum of the amount visible due to macro-terrain and the amount visible due to micro-terrain.

2.2.6 Determination of Line of Sight

There are two parts to determining if line of sight exists between an observer and a target. First, the target must be in the search sector for one of the observer's sensors. Each sensor has an azimuth that defines the center of its field of search (FOS) along with an angular sector width. Together these define the sector within which a target must appear in order to be seen. Secondly, the terrain between the observer and a target in its FOS must not block the line of sight.

A platform may have two sensors: a hunter and a killer. Line of sight to a target must be determined first. Whether the target is then in FOS must then be determined independently for each sensor since they may be searching in different directions or may have search sectors of different sizes. If LOS exists, acquisitions of the target must be scheduled independently for each sensor with the target in FOS.

Line of sight between each pair of platforms is updated each time one of them moves. A platform moves in steps based on its speed and a movement interval that applies to all platforms. The interval is divided by the platform's speed to determine the time required to move the interval. An event is scheduled at that time to update the platform's coordinates. Then, the list of other platforms in the moving platform's hunter or killer field of search is updated. If a platform has moved out of a sensor's field of search, acquisition of that target is canceled. If a platform has moved into the FOS of a sensor, a new acquisition of that target is scheduled. If a platform has remained in the FOS of a sensor, the acquisition is rescheduled to take into account the new distance between them.

Since the platform's movement may have caused it to move into or out of other platforms' fields of search, LOS and acquisition times for all other platforms looking at the platform that moved are also updated.

The steps that must be taken to determine if an observer (either the hunter or the killer sensor) has LOS to a target, and if so, the fraction of the target that is visible, are as follows:

- o Determine if the target is in the sensor's field of search. If not, stop processing for that sensor.
- o Through the use of the macro-terrain, determine the fraction of the target visible.
- o If micro-terrain is being used, determine the fraction of the target visible due to its being in defilade.
- o Determine the minimum fraction visible due to either the macro- or micro-terrain and return this value.

Figure 2-4 shows a typical battlefield with four blue platforms and thirteen red platforms. For one platform, HK2, the field of search (FOS) and the horizontal field of view (HFOV) is shown for both the hunter and killer sensors.

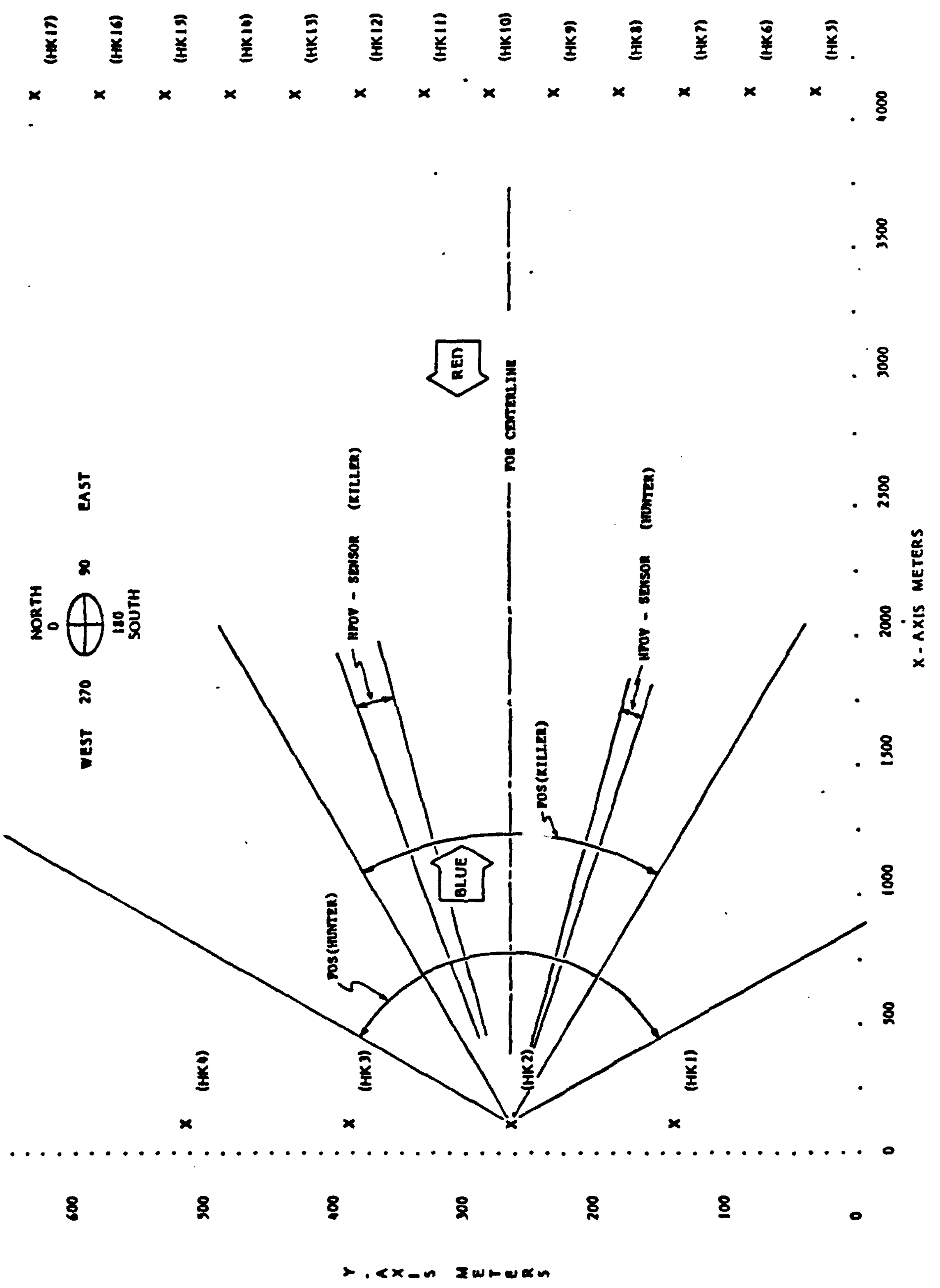

Figure 2-4, The Hunter-Killer Battlefield

2.3 PLATFORMS

The basic element of the Hunter-Killer Model is the platform, which represents a piece of equipment. This equipment may be an armed platform, such as a tank or an APC, that can detect and fire upon targets and be fired upon itself. A platform may also represent a crew-served weapon such as a TOW. It can also be, in the context of this model, primarily a target. Trucks and other large items that may be fired upon but have limited or no capability to return fire fall into this category. Platforms may also be airborne, such as helicopters.

Some equipment characteristics are the same for all pieces of that type of equipment. These characteristics include height, cross-sectional area, ground or airborne, and side (blue or red). A data structure called a type platform has been created to describe all common characteristics.

Other characteristics must be specific to a platform. These include initial coordinates, search sectors, and movement vectors.

These two sets of characteristics are discussed separately in the following sections.

2.3.1 Type Platform Characteristics

A type platform is the term used in the Hunter-Killer Model to refer to the generic characteristics of a model of military combat equipment, such as an M-1 tank or a T-72 tank. Only one offensive characteristic is associated with a type platform: the weapon it uses. All other characteristics are used to describe a platform as a target, as shown below:

Characteristic	Use
height	detection by thermal sensors
delta T	detection by thermal sensors
contrast	detection by contrast sensors
cross-section	laser rangefinding
reflectance	laser rangefinding

Type platforms are also used to describe weapon performance. The input data set contains a probability of kill curve for each type of weapon being modeled versus each type platform. See Section 2.6.

When micro-terrain is used, the fraction of a target not masked by terrain features (defilade) is given by a set of curves. One curve is input for each sensor class (ground or air) looking at each type platform. See Section 2.2.5.

2.3.2 Platform Characteristics

Whereas a type platform represents an equipment model, such as an M-1 tank, a platform represents a specific piece of equipment. The characteristics described by a type platform apply to all pieces of that model of equipment, and consequently are mostly target characteristics. The characteristics describing each individual platform control how that platform moves and acquires targets, and consequently are mostly offensive characteristics.

A platform is divided into two functional components: a hunter and a killer. Each component consists of a sensor and a set of rules governing how it attempts to acquire targets and what actions it takes once an acquisition is made. The hunter's function is to search for targets. Once the hunter has acquired a target it passes that target to the killer component to be engaged. The killer has two functions. When it is not engaging a target, it is searching in the same manner as the hunter. When it acquires a target, or when the hunter acquires a target and passes it to the killer, the killer stops searching and starts engaging the target. The killer continues to fire on the target until the target is dead, out of range, or line of sight is lost. While the killer is firing on the target, the hunter continues to acquire targets for the killer to engage.

Ground platforms do not communicate with each other, so a target found by one platform on a side cannot be passed to another platform. This means that a platform may have a killer only, which acquires and engages targets. If the platform has a hunter only, however, the platform will acquire targets but will be unable to do anything about them.

Flight teams, which consist of several airborne platforms acting together, do communicate with each other. The scout in a flight team may have a hunter only, which passes targets to another platform that has a weapon. By convention, the model requires a platform with only one component to have a killer. If the platform does not have a weapon this component then acts as a scout's hunter.

The characteristics used to describe a component are:

> The sensor: the ID of the sensor being used.
>
> The search sector: the horizontal and vertical width of the component's field of search.
>
> The search azimuth: the center line of the search sector.
>
> The Johnson level: the level to which a target must be acquired before the component takes action. The hunter's action is to pass the target to the killer. The killer's action is to engage the target. The Johnson level is described further in Section 2.9.
>
> The laser rangefinder: the ID of the laser being used. The rangefinder is only used by the killer component when firing on a target. The hunter has an input laser rangefinder to make the two components symmetrical, but it is not used.

These characteristics are described further in the discussion on Target Acquisition, Section 2.11.

2.4 MOVEMENT

2.4.1 Movement Of Ground Platforms

Movement of platforms is controlled through a movement interval and through movement vectors. The movement interval is an input value, typically set to 100 meters, that is used for all ground platforms. It is the size of the step used to update the coordinates of a platform. The movement vectors are a set of straight line segments along which a platform moves. Each vector is specified by an azimuth, length, and speed. Each platform may be given its own set of vectors. If a platform does not have a set of vectors, it does not move.

A platform begins each replication at its input initial X and Y coordinates. The speed for the first vector in its movement set is then used to compute the amount of time required to move the movement interval distance. An event is scheduled to occur at that time. When the event occurs it updates the platform's coordinates, computes the time to move another interval, and schedules another event. This continues until the platform has moved the entire length of the first vector. It then switches to the second vector in the set, which may have a different speed and azimuth. After the platform has moved along all vectors in its set, it stops and remains at that final position for the remainder of the simulation.

The speed of a platform should often be much less than the platform would actually move. The Hunter-Killer Model only moves platforms along straight line segments. On an actual road or battlefield, the movement is more complex. Also, actual speeds are a function of terrain and opposing force strength, both of which are ignored in this model. By reducing the speed the platforms move, the amount of time used to move across the battlefield more accurately represents realistic conditions.

When a platform moves, the model must determine which other platforms are in its search sectors, must determine if line of sight exists to those other platforms, and must schedule target acquisitions. This is discussed further in Sections 2.2 and 2.11.

2.4.2 Fixed vs Variable Search Azimuths

The direction in which a sensor searches may be a function of its movement vectors. At the start of a replication each sensor's azimuth is reset to its input initial azimuth. If the control FIXED.AZIMUTH has been set to YES, the sensor uses that same azimuth for the entire replication.

If FIXED.AZIMUTH has been set to NO, a sensor's azimuth changes each time its platform switches to a new movement vector The sensor's initial azimuth is read from the data set. At the time that value is read, the offset between that initial azimuth and the azimuth of the platform's first movement vector is computed. Then, each time the platform switches vectors, the sensor's azimuth is reset to keep the offset to the new movement vector constant.

The FIXED.AZIMUTH switch affects all sensors on all platforms in the simulation.

2.4.3 Movement of Airborne Platforms

An airborne platform moves in essentially the same manner as a ground platform, using a set of movement vectors; however, each vector has an altitude in addition to the speed, azimuth, and distance. When high resolution terrain is used, the altitude of the platform, when unmasked, is the elevation of the terrain plus the altitude given by the movement vector. If low resolution terrain is used, this altitude has no effect. See Section 2.2 for a discussion on the determination of line of sight.

An airborne platform may also fly in a loop, as when patrolling the FLOT. When constructing the movement vector set for an airborne platform, the keyword GOTO n is used to end the vector set, where n is the ID number of one of the vectors in the set. After the platform has moved to the end of the vector set, it starts using vector n and continues to the end of the set again, then cycles to vector n again.

Airborne platforms may move masked or unmasked. Airborne platforms may also be grouped into flight teams. In this case only the team leader has a movement vector set and all team members move together. Movement is discussed further in Section 2.5, Air Tactics.

2.5 AIR TACTICS

2.5.1 Airborne Platforms

The Hunter-Killer Model is capable of simulating airborne platforms in addition to ground platforms. An airborne platform, in terms of the Hunter-Killer Model, is simply another platform that is in the air at a given altitude instead of being on the ground. It may have one or two sensors, may have a weapon, may acquire targets, and may engage acquired targets. Also, it may become a target and be engaged or killed just like any other platform. The principal effect of the altitude is on line of sight. A masked platform cannot see targets and cannot be seen. When low resolution terrain is used, an unmasked platform can see 100% of the targets it detects (unless micro-terrain is used) and is 100% exposed to sensors able to see it. When high resolution terrain is used, line of sight and percentage visible to and from the platform are computed in the same way as for other platforms, except the altitude is added to the elevation of the ground at the platform's location. This has the effect of making targets more exposed than they are to a ground observer, but less exposed than with the constant 100% fraction visible of low resolution terrain. This is also a more accurate reflection of the difficulty of finding targets among the terrain features.

The use of airborne platforms must be thought of in terms of the model's design to study sensor performance. For example, three dimensional air-to-air combat is not simulated; instead, two airborne platforms attack each other as if they were two tanks. As a consequence, a scenario that includes many airborne platforms and has air-to-air engagements would give results that do not accurately reflect such combat. In terms of the Hunter-Killer Model, the typical ground scenarios with about 17 platforms -- 4 blue and 13 red -- should be retained. Air support would be provided by one, or at most two, flight teams per side.

An airborne platform may be a scout helicopter, an attack helicopter, a remotely piloted vehicle (RPV), or other such relatively low speed flying device. High speed platforms, such as an A10-Thunderbolt jet aircraft, should not be included. Helicopters usually fly masked to a location, then unmask to acquire a target and fire, then remask and fly masked to a new location. Since platforms cannot acquire targets or be acquired while masked and do not move while unmasked, these popup tactics mean that the speed of the platform has no effect on acquisitions. High speed platforms that are always unmasked, however, cannot be handled properly by the current acquisition techniques. The search module of the model computes the time to acquire a target as a function of range. To take into account the changing range as targets move, each platform moves in intervals of a fixed, input distance, typically 100 meters for ground platforms. When a move occurs, the coordinates of the platform are updated and all scheduled acquisitions that have not occurred since the last move are discarded. New acquisitions are scheduled based on the new ranges between platforms. With a high speed platform, the time interval between moves of 100 meters may always be less than the computed acquisition times, meaning that acquisitions would always be discarded and rescheduled but would never occur. Lengthening the distance moved to more than 100 meters would increase the time required to move, but may cause the platform to advance in unrealistically long steps. Balancing a reasonable move interval size against the computed acquisition times is simple for low speed platforms such as tanks. For high speed platforms, however, the limits of the model's design will be passed. High speed platforms may be included in the data set, and the model will run, but the results of the run may not be an accurate reflection of the high speed platform's performance or of the other platforms attempting to acquire it.

Low speed airborne platforms such as RPVs or helicopters flying at a fixed altitude should be given movement speeds much slower than they actually fly. In reality, these platforms fly winding paths using the terrain to mask themselves from the enemy while at the same time trying to see around terrain obstacles the enemy may be using to mask itself. This type of movement cannot be explicitly modeled by the Hunter-Killer Model; however, it can be represented by having the platforms move along their normal straight line paths at slower speeds than they actually fly.

2.5.2 Flight Teams

Unlike ground platforms, which operate independently of all other platforms, airborne platforms are grouped into flight teams (though a team may consist of a single platform). A team usually consists of two to four platforms. One platform has as its primary function the acquisition and designation of targets. RPVs and scout helicopters fall into this category. Whether this platform may also engage targets is determined through the data, which indicate which platforms have weapons. The additional team members are attack platforms. These may be airborne, such as helicopters, or ground, such as artillery. Based on current U.S. Army doctrine, it is assumed in this that the teams have one scout and multiple attack platforms; however, the actual composition of a flight team is determined by the user through the input data. The primary acquisition platform is referred to as the scout, whether it is capable of engaging or not. The other platforms are referred to as attack platforms.

When the members of a team are all airborne, the team flies as a group. If the attack platforms are ground based, they move (or remain stationary) independently of any action taken by the airborne portion of the team. The ground platforms move based on input movement vectors, just like ground platforms that are not supporting a flight team.

The scout performs all target acquisitions. If the scout has a weapon it also fires on the target. If the scout does not have a weapon it passes the targets to the attack platforms.

If the scout is killed, the next airborne platform in the team takes over the scout functions. If there are only ground platforms remaining, the team stops functioning. If all attack platforms are killed while the scout is still alive, and if the scout has no weapons allowing it to attack targets, the scout stops functioning and remains masked. No provisions are made for the scout to join another team or for other attackers to join the scout.

2.5.3 Flight Team Tactics

There are four tactics that may be employed by a flight team: FIXED, POPUP, ATR, and ATRMMW. These are set in the input data for each flight team. Each of these is discussed below.

2.5.3.1 FIXED Tactic

FIXED refers to flying at a fixed altitude. Exposure to detection is determined solely through terrain elevations, just as for ground platforms. This tactic would be used mainly by an RPV, which flies at an essentially constant altitude. If used by a helicopter that has a weapon and no other team members, that helicopter would function as if it were a tank in the air. This tactic may also be used to model a "sensor on a pole", an airborne sensor which does not move and is used for target acquisition and designation.

2.5.3.2 POPUP Tactic

POPUP refers to the most common tactic employed by helicopter teams. Using this tactic, the entire team flies together at all times (assuming all team members are airborne). When moving, the entire team is masked and consequently cannot become targets or acquire targets. The team flies masked for a fixed distance, set through the input data. Then, the scout performs a popup to acquire targets and request fire from the attack platforms.

If the scout popups up and acquires a target and if the target is in range of the weapons available to the flight team, a round is fired. The team then remasks. The scout then unmasks at the same position and tries to reacquire another target in range of its weapons. This continues until no targets are acquired or until all targets acquired are out of range. The team then remasks and flies to the next popup position.

On a battlefield the team would fire a round, change position enough so that the enemy would not know where it will appear the next time, popup again long enough to fire a single round, then move again. Since the platforms in this model do not remember where potential targets were located the last time they were seen, the small movements that teams make between rounds do not need to be explicitly modeled. In this model, popping up repeatedly in the same position does not increase the probability that the team will be acquired.

This tactic also assumes that the helicopters are using precision guided munitions or smart munitions. These weapons have a PK curve that is nearly flat for its entire range. As long as the target is within range, the team does not need to change position. Moving closer does not greatly affect the probability of killing an acquired target. This assumption conforms to current U.S. Army tactics which attempts to reduce helicopter vulnerability by giving them smart weapons that can be fired from 3 to 5 kilometers away from the target.

The procedure followed by the team once the popup position has been reached is as follows:

a) The scout unmasks to an altitude set in the input data.

b) For each platform in the scout's FOS to which it has LOS, the time to acquire the target is computed and a target acquisition is scheduled.

c) For each platform that has the scout in its FOS and also has LOS to the scout, the time to acquire the scout is computed and a target acquisition is scheduled. If the scout is acquired and killed before it remasks, the remainder of the team flies to the next popup location and starts with step a.

d) While the scout is unmasked, it is vulnerable to fire from any of the platforms from step c.

e) The scout remains unmasked until a target that is in range of the team's weapons has been acquired or until the maximum popup time (set in the input data) has been reached.

f) If the maximum popup time has been reached without a target having been acquired, the scout remasks. It then flys along its path to the next popup position and starts with step a.

g) If a target is acquired before the maximum popup time has passed, an engagements starts.

h) If the scout has weapons, the other team members remain masked.

i) If the scout does not have weapons, the attack platform must fire. If the weapon's target designator is the FIRER (see Section 2.6), it must first unmask and have the target passed to it by the scout. Once the attack platform unmasks, it can be acquired and killed by other platforms.

   It is assumed that communications between the scout and attack platform are sufficiently precise that when it unmasks the attacker will have the target in its narrow field of view.

j) A round is fired.

k) If the target designator does not need to remain unmasked until the round reaches the target, both the scout and the attacker mask. Otherwise the one that is not the designator masks.

l) After the round has reached the target, the model determines if a kill was made. If the target designator is still unmasked, it remasks.

m) The procedure loops to step a until no targets are within range, at which point the team flies masked to the next popup position.

2.5.3.3 ATR (Automatic Target Recognition) Tactic

A team using the ATR tactic performs almost identically to a team using the popup tactic. This team has an ATR sensor that changes the target acquisition procedure used once a popup position has been reached. It is assumed that every airborne member of the team has an ATR sensor. This allows another platform to take over the scout's job if the scout is killed without changing the tactics for the team.

The POPUP procedure is modified by replacing steps e, f, and g in Section 2.5.3.2 with the following:

e) The scout remains unmasked for an input ATR scan time, approximately 10 seconds.

f) The scout then remasks and processes the collected data. If targets were acquired, one is selected to be fired upon. Otherwise the team flies to its next popup position.

g) The scout unmasks and reacquires the selected target. It is assumed that the ATR sensor has the capability of having the selected target in its narrow FOV when the scout unmasks.

The POPUP procedure is then followed for steps h through l.

ATR sensors are discussed in more detail in Section 2.11.

2.5.3.4 ATRMMW Tactic

ATRMMW indicates that the flight team uses ATR sensors coupled with millimeter wave radar. This increases the probability of acquiring targets during the above step f, but has no other effect. ATR sensors fused with millimeter wave radars are discussed in more detail in Section 2.11.

2.6 WEAPONS

Each type of platform in the Hunter-Killer Model may have one weapon. Weapon performance is expressed through probability of kill curves which depend on the weapon being fired, the target's platform type, and the range to the target.

It has been assumed that platforms in the Hunter-Killer Model have no limitations on the number of rounds they can fire. The short duration of a Hunter-Killer scenario (30 to 60 minutes) combined with the type of combat portrayed serves to limit the rounds fired without an explicit control.

2.6.1 Probabilities of Kill

Weapons are modeled through probability of kill (PK) curves given in the input data. A curve may be defined for each weapon and type platform (target) combination. The input curve consists of a set of points, each of which gives a range and the probability of the weapon killing the type platform at that range. When a platform fires a round, the appropriate curve is selected. The range from the firing platform to the target is computed and the PK for that range is interpolated. A sample value between 0 and 1 is obtained using the SIMSCRIPT II.5 pseudo-random number generator. If the sample value is less than the PK, the target has been killed. If the sample value is greater than the PK, the round did not kill the target.

The above algorithm may be modified if the firing platform has a laser rangefinder. Range errors are one of the major factors in not successfully making a kill. By including a laser rangefinder on the firing platform, the probability of killing the target can be significantly increased. To allow for rangefinders, each weapon and type platform combination actually has two PK curves, one used when rangefinding is available and the other used when it is not. When a round is fired the firing platform uses its rangefinder (see Section 2.7). The success of the range finding as determined by the strength of the signal returned from the target controls the selection of the PK curve.

To reduce the amount of computer time used by the model, the laser rangefinder is used only once at the beginning of an engagement, regardless of how many rounds are fired at the target. The success or failure of the first use of the rangefinder determines the PK curve that is used for the entire engagement. In the Hunter-Killer Model the range between the firer and the target does not change significantly during an engagement and the short distance maneuvering that occurs on the battlefield is not modeled; consequently, more frequent checking of range would not change the results of an engagement.

An additional modification to the algorithm may be made if the weapon requires a target designator and if that designator is killed before the round reaches the target. This is described in the next section.

If a weapon does not have a PK curve for a particular platform type, that weapon cannot kill targets of that type.

2.6.2 Target Designation

Weapons may have one of three classes of target designation. These classes are specified in the input data for each weapon by the keywords NONE, FIRER, and SCOUT. The classes are interpreted as follows:

> **NONE.** This applies to the majority of munitions, those that do not require a target designator. Once fired a round travels to the target without further direction from an external source. This class includes smart munitions such as those with internal heat seeking systems. Smart munitions are distinguished from other munitions in this class by having higher probabilities of kill.
>
> **FIRER.** This applies to munitions that must be guided to the target by the same platform that fires them. This includes TOWs and all wire guided munitions.
>
> **SCOUT.** This applies to munitions that must be guided to the target, but where the target designator may be separate from the firing platform. This includes artillery shells such as Copperheads and all laser guided munitions such as Hellfire. These may be guided to the target by scout helicopters, RPVs, or other types of forward observer. Only in the case of a flight team may the firing platform and the target designator be separate. This situation is discussed in Section 2.5.

The target designators are not explicitly modeled. Instead, if a weapon requires a designator, it is assumed that the designator is available.

For guided munitions (SCOUT and FIRER), the platform performing target designation must survive until the round reaches the target. If the designator is killed before the round arrives, the PK for that weapon is degraded by an input curve that relates a degradation factor to the distance from the target to the round at the time the designator was killed. This means that if the designator was killed while the round was still distant from the target, the PK may drop to zero. If the round has almost reached the target, the PK may be degraded only slightly.

2.6.3 Miscellaneous Weapon Characteristics

Each weapon has a reload time, set through the input data, which may be 5 to 10 seconds. This value does not need to be the actual time required to reload the weapon; instead, it is intended to provide a throttle on the firing rate of a platform. With a reload time of zero a platform would fire a round, assess the effects, and continue firing until the target was killed. All of this would occur without any simulation time passing. Since no simulation time was passing, no other events could occur, such as other platforms firing or other sensors acquiring targets. A reload time greater than zero ensures that after a platform has fired a single round, other platform may act -- possibly killing the first platform -- before the first platform fires another round.

Each weapon also has an input round velocity. As described in Section 2.6.2, when the weapon requires a target designator, the designating platform must survive until the round reaches the target, otherwise the PK is degraded. This input velocity along with the distance to the target determines the length of time the designator must survive. The interval between shots is the maximum of the reload time and the time required for the round to reach the target. This value is only meaningful if the target designator is FIRER or SCOUT. If the designator is NONE, this value is ignored (and may be set to zero in the data) and the firing rate is determined solely through the reload time.

2.7 LASER RANGEFINDERS

Lasers are used in the Hunter-Killer Model only as rangefinders. As discussed in Section 2.6.2, some weapons may require laser target designators, but it is assumed that these are available when needed. No attempt is made to explicitly model a target designator by determining the reflected signal strength, whether the signal is strong enough for the round to home on it, and other such aspects of designators. It is assumed that these factors are included in a weapon's probability of kill.

Lasers must operate at a wavelength of 1.06 or 10.6 microns. The model only has equations to compute the atmospheric transmission of the signal at those wavelengths (see Section 2.10.3 for spectral bands 0.7 to 1.1 and 8 to 12 microns).

To determine the success of laser rangefinding, the received signal power must be computed. The following equation, taken from the *Visionics E-O Sensor Performance Handbook* and the *RCA Electro-Optics Handbook*, is for a pulsed rangefinder with matched filter detection, such as the Nd:YAG system currently used by the U.S. Army.

$$P_s = [P_t\ T_t]\ [T_a]\ [M\ r_t \cos(A)]\ [T_a]\ [T_r\ A_r\ /\ (\text{pi } R^{**}2)]$$

where

P_s = received signal power, watts
P_t = peak transmitter power, watts
T_t = transmitter optics transmittance, dimensionless
T_a = atmospheric transmittance between laser and target
M = ratio of target area to illuminated area, dimensionless
r_t = target reflectance, dimensionless
A = angle between a normal to the target and the beam
T_r = receiver optics transmittance, dimensionless
A_r = receiver effective aperture area, square km
R = range to the target, km.

Brackets are included in the above equation to separate out the terms for transmitted power, propagation of the laser radiation to the target, diffuse reflection of the laser radiation by the target, propagation from the target to the receiver, and interception of the reflected signal at the laser receiver.

The atmospheric transmission, T_a, is computed as:

$$T_a = T_A\ T_S$$

where

T_A = atmospheric transmission of infrared radiation from Section 2.10.3
T_S = smoke transmission from Section 2.10.10 (if smoke is modeled).

The factor M accounts for underfilling the target at short ranges and overfilling the target at long ranges. It is computed as:

$M = A_t / (pi R^{**}2 s^{**}2)$

where

s = half angle laser beam divergence, mrad
A_t = target area, square meters.

M must be equal to or less than 1.0.

As a simplification, the Hunter-Killer Model assumes that A -- the angle between the normal to the target and the laser's beam -- is zero in all cases.

After computing the received signal power, the signal to noise ratio is computed as:

$SNR = P_s / NEP$

where

SNR = signal to noise ratio, dimensionless
NEP = detector noise equivalent power, watts.

The rangefinder's threshold to noise ratio is computed using the following equation which is based on the *RCA Electro-Optics Handbook*:

TNR = SQRT[-2.0 ln(2.0 SQRT(3) PW FAR)]

where

TNR = threshold to noise ratio, dimensionless
PW = pulse width, seconds
FAR = false alarm rate, number per second

and SQRT(x) indicates the square root of x.

If the computed STN is greater than the laser's TNR, the rangefinding was successful.

2.8 SENSORS

2.8.1 Sensor Characteristics

The Hunter-Killer Model is capable of simulating a large variety of sensors, both contrast and infrared. Target resolution by a contrast sensor is by minimum resolvable contrast (MRC). Target resolution by an infrared sensor is by minimum resolvable temperature (MRT). In addition, infrared sensors may perform minimum detectable temperature (MDT) acquisitions. These occur when the sensor can perceive, but not necessarily resolve, a target. In such a situation, the observer may perceive a hot spot, but be unable to resolve the source.

The Hunter-Killer Model divides sensors into operating spectra as follows:

Spectrum (microns)	Resolution Technique	Sensor type
0.4 - 0.7	MRC	Direct view optics
0.4 - 0.9	MRC	2nd generation II
0.6 - 0.9	MRC	3rd generation II, Si TV
0.7 - 1.1	MRT	near infrared
3.0 - 5.0	MRT	middle infrared
8.0 - 12.0	MRT	far infrared

In the Hunter-Killer Model, a sensor's spectrum is used to determine which technique is used to compute the transmission of a signal from the target to the sensor.

A sensor has three values that define its field of view: the horizontal FOV width, the vertical FOV width, and the FOV scaling factor. The input horizontal and vertical FOV widths are for the sensor's wide FOV. These values are only used in computing the number of FOV in the sensor's FOS, which is then used to determine the time required to detect a target. The FOV scaling factor is used in conjunction with the resolution curves and is defined below in Section 2.8.3.

When a sensor's field of view is changed between narrow and wide, there is a small time delay before the new image can be used. This delay is caused by hardware effects plus the time required by the operator to refocus on the target. Each sensor has an input value that represents this delay time, typically around 5 seconds.

2.8.2 Automatic Target Recognition (ATR) Systems

Helicopters are highly vulnerable when unmasked. As a consequence, sensor systems are being developed which will minimize the amount of time helicopters are exposed. These automatic target recognition (ATR) systems allow a helicopter to unmask for a brief amount of time, typically for ten seconds, while the system scans the field of search and stores the imagery. The helicopter then remasks and hovers while the system postprocesses the stored imagery.

The helicopter cannot remain masked for too long while the operator reviews the stored imagery or the targets will have move sufficiently to make reacquisition difficult. An ATR system is capable of cuing the operator to likely targets so that the operator does not need to review the entire stored FOS. As the operator reviews the targets a priority is assigned to each. The helicopter then unmasks and fires. The ATR system uses the stored target priorities and locations for rapid reacquisition of the targets. The information can also be passed off to other members of a flight team for action.

These systems can also be used in a manual mode like other sensors.

The Hunter-Killer Model simulates the use of ATR systems by flight teams. It is assumed that these systems will all be FLIRs.

An ATR system requires two values in addition to those used by other sensors, a signal to noise factor and a pixels on target factor. These are discussed further in Section 2.11.

ATR systems have a high false alarm rate when operating in the automatic mode. The Hunter-Killer Model simulates the fusion of an ATR system with a millimeter wave radar to increase the probability of detection. This is also discussed in Section 2.11.

2.8.3 Resolution Curves

Resolution curves are used to determine the spatial frequency of a target. Each curve is defined by exactly twenty points, with linear interpolation used to compute values between the input points. If a given value is outside of the domain of the curve, the appropriate end point of the curve is used and a warning message is printed.

The values given by a resolution curve depend on the sensor's resolution technique. For a contrast sensor, each resolution curve gives the spatial frequency of the target, in cycles per milliradian, as a function of the target's apparent contrast. For a thermal sensor, each curve gives the spatial frequency of the target as a function of the detected temperature difference between the target and the background, in degrees Celsius.

Each sensor may have one or more resolution curve. A sensor may have one curve to be used for narrow FOV and a different curve for wide FOV. If a sensor has only one curve, the FOV scaling factor is used to determine the spatial frequency when the sensor is in the mode opposite that of the curve. If the curve is for narrow FOV and the sensor is in wide FOV, the computed frequency is modified as follows:

F_wide = F_narrow / scaling

If the curve is for wide FOV and the sensor is in narrow FOV, the computed frequency is modified as follows:

F_narrow = F_wide * scaling

Note that since narrow FOV always improves a sensor's performance -- measured by an increased spatial frequency -- the scaling factor must be greater than or equal to one. If a sensor has only one resolution curve and if its scaling factor is 1.0, then the sensor has only one field of view available.

A contrast sensor may have more than one curve each for narrow and wide FOV. The resolution curves for a contrast sensor are based on a specific light level, measured in foot-candles. If the battlefield's light level, set in the Controls section of the data set, is not the same as the light level for available resolution curves, two curves must be input, one for a lower light level and one for a higher level. The target's spatial frequency is computed using each curve. The final spatial frequency is interpolated between the two values.

Thermal sensors that can perform MDT acquisitions have an additional curve giving the minimum detectable temperature, in degrees Celsius, as a function of the inverse of the target's height, in 1/milliradian. Only one MDT curve, for narrow FOV, is used.

Further discussion on the use of these curves appears in Section 2.10.

2.9 JOHNSON CRITERIA

The Johnson Criteria are a set of four values used to define levels of target discrimination. The levels, in order of increasingly precise discrimination, are: detection, classification, recognition, and identification. These level may be defined as follows:

Detection (level 1) is the ability to distinguish that an object is of military interest.

Classification (level 2) is the ability to distinguish a target by general type; e.g., as a tracked vehicle instead of a wheeled vehicle.

Recognition (level 3) is the ability to distinguish between two targets of similar type; e.g. between two type of tracked vehicles such as APCs and tanks.

Identification (level 4) is the ability to discriminate the exact model of a target; e.g., T72 or M1 tank.

Each level is defined by the minimum number of cycles from a standard bar type target pattern that must be distinguished for a target to be acquired at that level. Figure 2-5 presents the discrimination levels as a function of a number of cycles.

The Johnson levels are used to construct a target hierarchy in the Type Platform section of the data set. The type platforms described in Section 2.3 are all level 4 targets. The hierarchy relates these level 4 targets to level 1, 2, and 3 targets. A typical hierarchy is shown in Figure 2-6.

As shown in Figure 2-6, a component may acquire a T72 tank at a Johnson level of 3. This means that the component can distinguish the target as a tank of some kind but not the specific type of tank. If the target is acquired at a level of 2, the component can only determine that the target is some kind of tracked vehicle. If the target is acquired at a level of 1, the component can only determine that a target is present.

The Hunter-Killer Model uses these criteria for several purposes. When an acquisition occurs, the level to which the target can be discriminated is determined. The next action taken by the acquirer is based on that level. Each component of a platform has an input minimum action level to which a target must be acquired before the component takes action against the target. For a hunter component, this action is to pass the target to the killer component. For a killer component, this action is to fire on the target. If the target is acquired at a level lower than the action level it will be ignored.

Also, the time required to acquire a target is dependent on the action level. A component can quickly determine that some type of target is on the battle-field. Acquisition takes much longer if the component is required to inter-rogate the target long enough to determine the specific type of target. See Section 2.10.7.

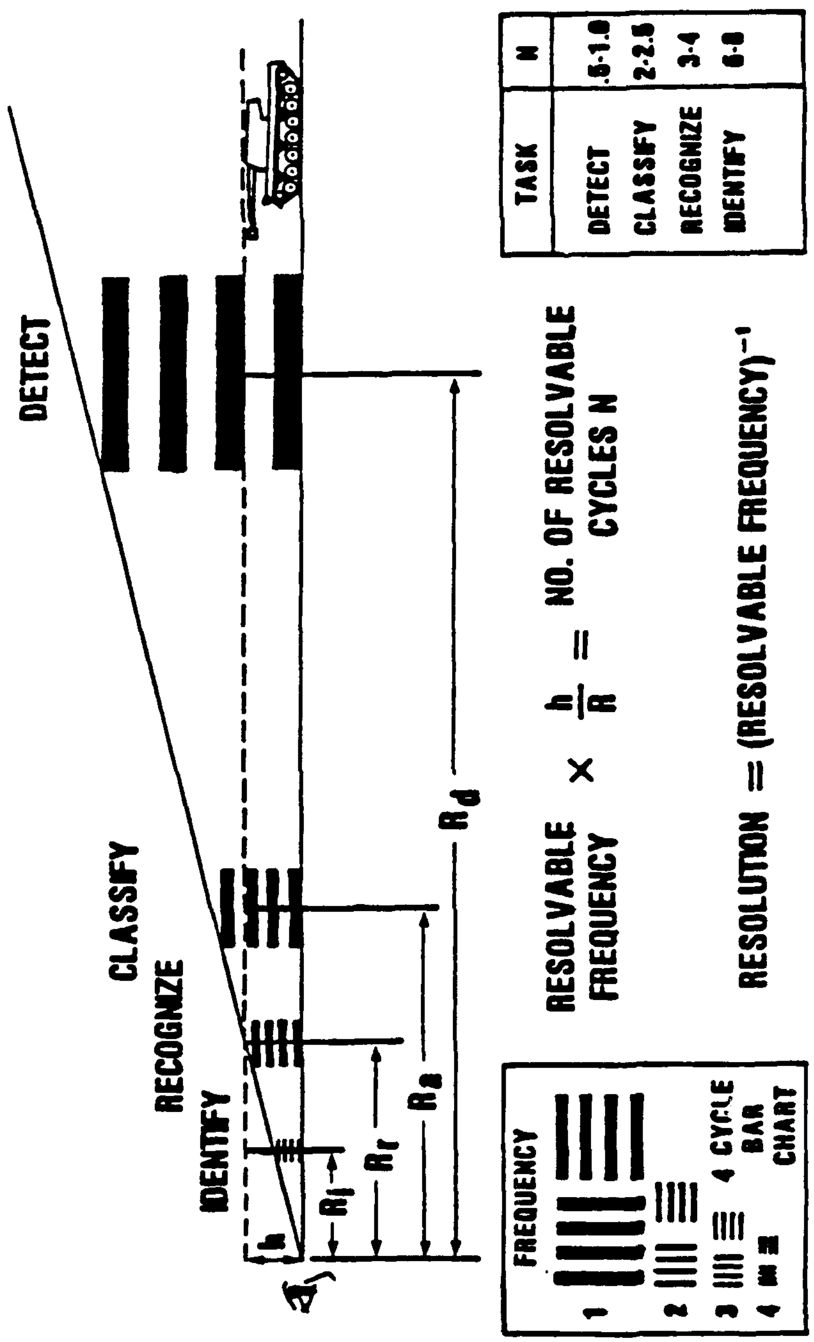

Figure 2-5, Johnson Resolution Criteria

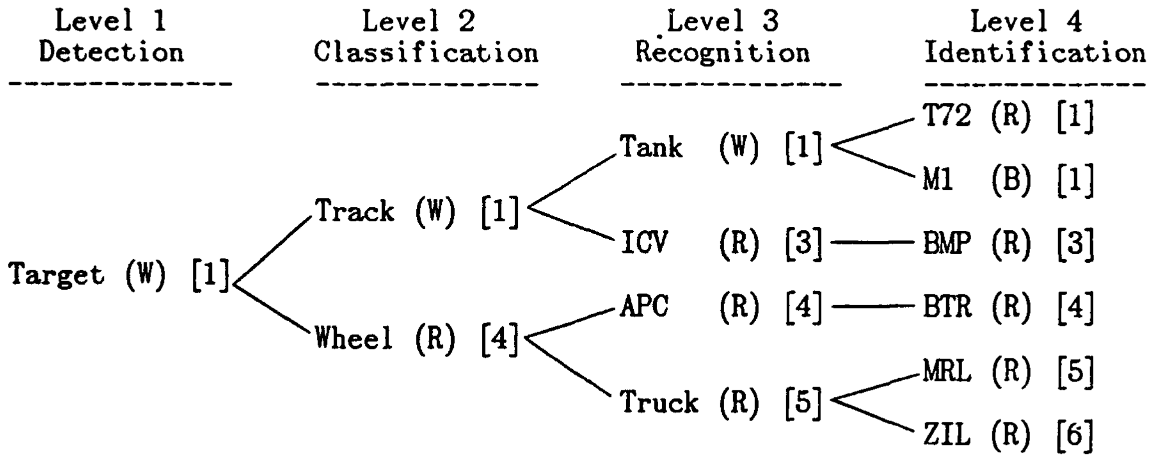

Figure 2-6, Johnson Hierarchy of Targets

In Figure 2-6, the letter in parentheses after the target name is the side to which it belongs: B means blue, R means red, W means white (neither blue nor red). The level 3 target called tank is white because there are two types of tank on the battlefield, the T72 and the M1. At the recognition level 3, an observer cannot distinguish between them. A killer component with an action level of 1, 2, or 3 which acquired the target at level 3 would fire on it even though the target may be friendly.

The level 3 target ICV (infantry combat vehicle) is red because the only level 4 ICV on the battlefield is the red BMP. Similarly, the level 3 targets APC (armored personnel carrier) and truck are red because all targets lower in the hierarchy are all red.

The data set assigns priorities to targets in a similar manner. Priorities are defined for level 4 targets, shown in square brackets in Figure 2-6. A lower number indicates a higher priority, with 1 being the highest priority. These priorities are used by the killer to decide which target to engage first if the platform has detected multiple targets at the same time.

A level 1, 2, or 3 target is assigned the highest priority of the targets under it in the hierarchy. The level 3 tank has a priority of 1 because both of the level 4 targets below it are level 1. The level 2 track target also has a priority of 1, the highest of the priorities of the level 3 targets beneath it -- tank and ICV.

2.10 ACQUISITION TIMES AND LEVELS

This section describes the method of computing the time required to acquire a target and the Johnson level at which it will be acquired. Use of these values is discussed in Section 2.11, The Simulation.

2.10.1 The Search Model

The procedure to compute acquisition times and Johnson levels is based on the NV&EOC Search Model. The Search Model is a single target, single observer model consisting of display and physical search submodels. The display submodel predicts probability of acquisition versus time for a fixed viewing device. Observers acquire a target by visually searching the device display with eye movements. The physical search submodel extends the display model to predict the probability of acquisition when the observer must scan the device over a known search field in order to acquire the target.

The major assumptions and limitations contained in the Search Model are listed below.

The target is within the sensor's field of search and line of sight exists.

The observer acquires a target only by its primary signature, IR or contrast. The target's location, motion, and weapon firing have no effect on acquisition.

The target is static and does not make and break LOS nor does the range vary during the attempted acquisition.

The FOS search tactics are an unspecified, uniform scan.

Ground targets have average and unspecified background clutter.

The atmosphere is homogeneous.

All observers are equally good (or bad) at the skills of target acquisition. The performance of observers does not vary over time; i.e., they do not experience learning or fatigue.

The observers do not have memory. If a target is acquired at too low a level to take action the observers do not file the target into their memory for later interrogation.

The sensors make Class I errors in that they fail to acquire legitimate targets. The sensors do not make Class II errors in that they never acquire an object that is not a target. Even if false targets are correctly rejected, the interrogation process would consume time that could be spent searching.

Each target is in a separate FOV and is acquired independent of all other targets. The sensor does not acquire the second and subsequent targets in a unit based upon acquiring the first.

The scan rate is independent of the environment or number of acquisitions occurring. The sensor does not interrogate regions that have previously produced acquisitions. The scan rate continues without regard to the visibility.

All targets are the same except for critical dimension and temperature; for example, tanks are not assigned unique and observable properties to distinguish them from motor-generator sets.

Sensors perform at full capability and do not experience failure (partial or complete) from either normal failure or battle damage. Sensors can perform until their platform is killed.

Sensors do not communicate and cooperate in search with the exception of hunter/killer pairs and flight teams.

There is no distinction between night and day except through data such as the resolution curves or the background light level used by contrast sensors. During daytime the sun position has no effect.

Masking by vegetation is not considered, except as micro-terrain is used to reduce the fraction of the target's critical dimension that is visible to the observer.

The steps used by the Search Model to compute the acquisition time and level are discussed in the following subsections.

2.10.2 Spatial Frequency - Contrast Sensors

The Hunter-Killer Model defines three spectral bands for contrast sensors: 0.4 to 0.7, 0.4 to 0.9, and 0.6 to 0.9 micron. The computations that follow are dependent on the sensor's band.

The first step in determining the target's spatial frequency is to compute the target's apparent contrast, which is the target-to-background contrast seen by an observer separated from the target by a contrast degrading medium.

The visible extinction coefficient is obtained from Koschmieder's equation:

$$s_{.55} = - \ln(0.02) / V$$

where

V = visibility, km

and ln(x) is the natural logarithm of x.

The atmospheric extinction coefficients, E, used by the Hunter-Killer Model for visible light are as follows:

$E = s_{.55}$, for sensors operating at 0.4 to 0.7 microns

$E = s_{.55}\ (0.75 / 0.81)$, for sensors operating at 0.4 to 0.9 microns

$E = s_{.55}\ (0.59 / 0.81)$, for sensors operating at 0.6 to 0.9 microns.

The contrast transmission, T_c, is then:

$$T_c = \frac{1}{1 + S\ [\ (1/T) - 1\ ]}$$

where

S = sky to ground ratio for the sensor's spectral band, dimensionless
T = $T_E\ T_S$
T_E = e **(-E)
T_S = smoke transmission factor from Section 2.10.10.

The apparent contrast, C_a, is given by:

$C_a = C_t\ (T_c$ ** R)

where

C_t = input target contrast
R = range to target, km.

The target's spatial frequency is then interpolated from the sensor's twenty point resolution curve. For contrast sensors, these curves were obtained at specified light levels. If the battlefield light level does not equal the light level for the curves available for the sensor, two curves to bracket the battlefield light level must be input. The spatial frequency is then obtained from each curve. The target's spatial frequency is then interpolated between those two values.

The resolution curves are obtained by measuring sensor performance while in either narrow or wide FOV. If the sensor has a resolution curve for its current FOV, that curve is used. If no curve is available the curve for the opposite FOV is used and the frequency is scaled. See Section 2.8.3.

2.10.3 Spatial Frequency - Thermal Sensors

The Hunter-Killer Model defines three spectral bands for thermal sensors: 0.7 to 1.1, 3 to 5 and 8 to 12 microns. The computations that follow are dependent on the sensor's band.

The difference between the target's temperature and the background temperature, as detected by a sensor, must be computed first.

The atmospheric transmission of the infrared radiation after being attenuated by molecular absorption, T_A, is:

T_A = e ** (-A R)

where

A = absorption factor, 1/km
R = range to the target, km.

The model reads a separate absorption factor for each of the operating spectra.

The atmospheric extinction coefficients, E, which account for scattering and absorption by suspended aerosol particles are as follows. These have been taken from the NV&EOC's G/AP (Grafenwoehr / A.P. Hill) Model for combined wet/dry fog.

For sensors operating at 0.7 to 1.1 microns:

$$E = \text{minimum}(E1, E2)$$

where

$$E1 = 10 ** \{-0.14 + [1.16 \log(s_{.55})]\}$$

$$E2 = 10 ** \{\log(s_{.55}\}$$

For sensors operating at 3 to 5 microns:

$$E = 0.0 \quad , s_{.55} < 0.98$$

$$E = 20.0 \quad , s_{.55} > 7.82$$

$$E = 10 ** \{-1 + 2.404 \log(s_{.55}) - 0.511 [\log(s_{.55}) ** 2] \} ,$$

$$7.82 <= s_{.55} <= .98$$

For sensors operating at 8 to 12 microns:

$$E = 0.14 \quad , s_{.55} < 0.98$$

$$E = 20.0 \quad , s_{.55} > 7.82$$

$$E = 0.14 + 10 ** \{-0.980 + 1.851 \log(s_{.55}) - 0.212 [\log(s_{.55}) ** 2] \} ,$$

$$0.98 < s_{.55} < 7.92$$

where

$s_{.55}$ = visible extinction coefficient from Section 2.10.2

and log(x) is the base 10 logarithm of x.

The transmission of the infrared radiation after being attenuated by the G/AP effect is:

$$T_E = e ** (-E R).$$

If smoke is modeled, the transmission through smoke is T_S as computed in Section 2.10.10.

The temperature difference detected by the sensor, dT', combines these factors:

$$dT' = T_A \, T_E \, T_S \, dT$$

where

dT = temperature difference between the target and the background as measured at the target, degrees Celsius.

The target's spatial frequency in cycles per milliradian is then interpolated from the sensor's twenty point resolution curve using dT'.

The resolution curves are obtained by measuring sensor performance while in either narrow or wide FOV. If the sensor has a resolution curve for its current FOV, that curve is used. If no curve is available the curve for the opposite FOV is used and the frequency is scaled. See Section 2.8.3.

2.10.4 Resolvable Cycles

The number of resolvable cycles across the target, NRC, is computed as follows:

$$NRC = F \, D \, / \, R$$

where

F = calculated spatial frequency of the target, cycles per milliradian
D = target critical dimension, meters
R = range to the target, km.

The target's critical dimension is the measurement that limits the ability to acquire the target. For most targets the critical dimension is the height. The critical dimension of a ground target is the amount of the target visible, not necessarily the full height of the target, see Section 2.2.

2.10.5 Acquisition Level

Each of the four Johnson levels discussed in Section 2.9 has an input value, N50, which represents an empirically established number of resolvable cycles across a target that yields a P_infinity of 50% for an observer trying to acquire a target at that Johnson level with a specified background clutter level. Different sets of values may be used for contrast and thermal sensors. Typical N50 values for medium clutter are:

Johnson Level	N50
Detection	1.0
Classification	2.0
Recognition	4.0
Identification	8.0

The NRC value from Section 2.10.4 is compared to the input N50 values for contrast or thermal sensors, as appropriate, to determine the acquisition level. If the NRC is less than N50 for detection, the target cannot be acquired.

2.10.6 P_infinity

P_infinity is the probability of acquiring a target when given an infinite amount of time in which to search. This value may also be interpreted as the fraction of a set of observers searching the same field who will acquire the target given an infinite amount of time in which to search.

P_infinity is computed as follows:

$$\text{let } E = 2.7 + 0.7\,(NRC / N50)$$

Then

$$P\_infinity = \frac{(NRC / N50) ** E}{1 + (NRC / N50) ** E}$$

where

N50 = N50 value for the Johnson level at which the target must be acquired before the acquiring platform takes action against the target.

2.10.7 Acquisition Time

The probability of detecting a target at a time T is estimated by:

$$P\_detect = P\_infinity \{1 - e ** [-T / (N\,T_a)]\}$$

where

N = number of sensor fields of view in the field of search
T_a = mean acquisition time, seconds.

The value of T_a has been empirically determined to be:

$$T_a = 3.4 / P\_infinity.$$

The time to acquire a target, in seconds, can be derived from this equation as follows:

$$T = - N T_a \ln[1 - (P\_detect / P\_infinity)]$$

where ln(x) is the natural logarithm of x.

This equation has two unknown values, T and P_detect. The Hunter-Killer Model obtains a value between 0 and 1 for P_detect using the SIMSCRIPT II.5 pseudo-random number generator. If P_detect is greater than P_infinity, the acquisition does not occur. If P_detect is less than P_infinity, the above equation is used to compute the acquisition time.

It is possible for this calculated value to be unrealistically small. To prevent this, the Hunter-Killer Model uses the maximum of the calculated T and and of an input minimum acquisition time.

For the purposes of this model, an acquisition time greater than the input length of a replication is considered to be infinite. This definition of an infinite acquisition time is used in the collection of sensor performance statistics. As described in Section 4.4, the minimum, maximum, and average of the non-infinite acquisition times are collected for each sensor. In addition, the total number of infinite acquisitions is collected.

The value of N, the number of fields of view in the field of search, is always the number of wide FOVs in the FOS. The search algorithm for the Hunter-Killer Model causes a sensor to search in wide FOV only, assuming the sensor has both a wide and narrow FOV. Narrow FOV is only used to improve the resolution of an acquired target or, for the killer, to conduct an engagement. This means that acquisition times will only be computed when the sensor is in wide FOV.

2.10.8 MDT Acquisitions

Some thermal sensors are capable of perceiving a "hot spot" even if they cannot acquire a target. This is referred to as a minimum detectable temperature (MDT) acquisition. These computations provide the acquisition level, replacing Section 2.10.5, and the P_infinity, replacing Section 2.10.6.

A sensor capable of performing an MDT acquisition has, in addition to its MRT to spatial frequency curve, a twenty point curve giving the MDT in degrees Celsius as a function of the inverse target size, in 1/milliradians. If the computed NRC from Section 2.10.4 is less than the N50 number of cycles for a Johnson level of detection, the target cannot be acquired. If the sensor has an MDT resolution curve, the model determines if an MDT acquisition is possible.

The first step is to compute the perceived inverse target size, I, measured in 1/milliradian:

$$I = R / D$$

where

R = range to the target, km
D = target critical dimension, meters.

This value is used to interpolate the MDT from the resolution curve. If the MDT is less than 0.01, no acquisition occurs. If the MDT is greater than 0.01, the signal to noise, STN, is computed as:

STN = dT' / MDT

where

dT' = value from Section 2.10.3.

P_infinity is computed as:

X = 2.2274 + 2.4135 STN

P_infinity = X / (1 + X).

The acquisition level is level 1, detection.

2.10.9 BIFF

The above procedure may be modified by the use of Battlefield Identification - Friend of Foe. If BIFF is modeled, two values are read by the model: the probability of successful BIFF of a blue target by a red sensor and the probability of successful BIFF of a red target by a blue sensor.

If the computed acquisition level from Section 2.10.5 or 2.10.8 is at least detection, the sensor is capable of making an acquisition. The model then obtains a value between 0 and 1 from the SIMSCRIPT II.5 pseudo-random number generator. If that value is less than the appropriate input BIFF probability, then BIFF is considered to be successful and the acquisition level becomes the highest level, identification.

2.10.10 Smoke Effects

The amount of visible light or infrared radiation transmitted to the sensor can be significantly reduced by the presence of smoke on the battlefield. Through the Controls section of the data set, smoke can be included in the simulation. The Hunter-Killer Model uses a simple smoke submodel to compute an extinction factor that is then used in Sections 2.10.2 and 2.10.3.

Smoke is deployed ten seconds into a replication and expands at a rate of 50 meters per minute. The effects are for 24 rounds of white phosphorus in an unstable atmosphere, relative humidity of 80%, with a 1.5 meter per second quartering wind on a June afternoon at Grafenwoehr.

The smoke is homogeneous across the battlefield instead of being localized. This simplification is used since the Hunter-Killer Model is most appropriately employed for a small number of platforms with the platforms on each side located close to each other -- that is, an engagement between two units. In such a situation, smoke is deployed between the units across the entire FLOT separating them.

The transmission of a signal through smoke, T_S, is dependent on the wavelength of the signal. The transmission factor is computed as follows:

let D = the amount of time, in seconds, that smoke has been deployed.

For D equal to zero:

$$T_S = 1.0$$

For signals from 0.7 to 1.1 and from 3 to 5 microns, T_S is the minimum of 1.0 and:

$$T_S = 0.005081 \, [e ** (0.003689 \, D)] \qquad , D > 0$$

For signals from 8 to 12 microns:

$$T_S = 0.4 \, \{1.0 - [0.27114 \, D \, e**(-D / 10)]\} \qquad , D > 0$$

The above transmission factor is not dependent on range between the observer and the target. It is assumed that the smoke was deployed between the platforms and that the entire cloud separates them. When the platforms are close to each other, however, they will be inside the cloud itself. Since the entire cloud is no longer between them, the transmission factor must be increased.

It is assumed that the smoke has been deployed at a distance of 800 meters from the observer. If the range between the platforms is greater than 800 meters, the above transmission factor is used. Otherwise, the width of the cloud, in meters, is computed as:

$$W = 50 * (D / 60)$$

where 50 is the expansion rate of 50 meters per second. The distance from the deployment point to the target, in meters, is computed as:

$$P = 800 - R$$

where R = range from the observer to the target, meters.

This situation is shown in Figure 2-7. The letter O stands for the observer, T for the target, D for the deployment point, and S for the leading edge of the smoke cloud.

If $W < P$, the target is closer to the observer than the leading edge of the cloud. Consequently, there is no smoke to reduce transmission and T_S becomes 1.0.

If $W > P$, only a portion of the cloud is degrading transmission. The new transmission factor, T_S', is computed as the minimum of 1.0 and

$$T_S' = T_S \, [e ** -\ln(T_S \, P \, / \, W)].$$

where ln(x) indicates the natural logarithm of x.

```
<---------------- 800 meters ---------------->
                                  <---- W ---->
                                  S
                                  S
                                  S
                                  S
O                                 S  T        D
                                  S
                                  S
                                  S
                                  S
<--------------- R ----------------><-- P -->
```

Figure 2-7, Deployment of a Smoke Cloud

2.11 AUTOMATIC TARGET RECOGNITION SYSTEM ACQUISITIONS

2.11.1 ATR Overview

An ATR sensor may be used in automatic mode to allow a helicopter to quickly scan a FOS, then remask to examine the stored imagery. After targets are selected, the helicopter will unmask to fire. The same sensor will be used, but in manual mode. This section discusses acquisitions in the automatic mode. In the manual mode the system operates in the same way as other sensors. The system will be in narrow FOV when scanning in the automatic mode.

Two factors are used to determine the probability of detection when the system is in automatic mode. The first is a signal to noise (STN) factor and the second is a pixels on target (POT) factor. These factors are related to detection probabilities based on empirical data.

The system processes each target in its FOS to determine if a detection occurs and, if so, the Johnson level at which the target is acquired.

The system is capable of indicating probable targets to the operator. Through the data set the user determines the time required by the operator to examine each target. This interrogation time multiplied by the number of acquired targets determines how long the helicopter will remain masked.

The probability of detection by an ATR sensor in the automatic mode will not be discussed in detail. Only the equations used will be presented in the following subsections. Further details on the development of these equations may be found in *Advanced Processor/FLIR Performance Measures*, see Appendix A.

2.11.2 Signal to Noise Factor

The signal to noise factor, STN, is computed as follows:

$$STN = \frac{dT\ (t)^{**}R\ SQRT(N)}{SQRT(2)\ \ NETD}$$

where

dT = difference between the target and the background temperatures, degrees Celsius (see Section 2.10.2)
t = atmospheric transmittance, 1/km
R = range to the target, km
N = pixels combined to improve signal to noise.
NETD = noise equivalent temperature difference for actual electronic bandwidth.

and SQRT(x) indicates the square root of x.

The N and NETD terms are dependent on the specific ATR system. The S/N FACTOR term defined in the Sensors block of the data set (see Section 3.3.5) is computed as the constant portion of the above equation:

$$S/N\ FACTOR = \frac{SQRT(N)}{SQRT(2)\ NETD}$$

The curve relating STN to a detection probability, P_STN, is shown in Figure 2-8. The curve definition, based on empirical data, is:

$$
\begin{aligned}
P\_STN &= 0.0 && , STN < 3.0 \\
P\_STN &= 0.2\ STN - 0.6 && , 3.0 < STN < 7.0 \\
P\_STN &= 0.1\ STN + 0.1 && , 7.0 < STN < 9.0 \\
P\_STN &= 1.0 && , 9.0 < STN
\end{aligned}
$$

2.11.3 Pixels on Target Factor

The effective pixels on target factor (POT) is computed as:

$$POT = \frac{Dx\ Dy}{Yx\ Yy} \ \frac{L\ \ W}{N\ R^{**}2}$$

where

L	= target length, meters
W	= target width, meters
R	= range to the target, km
N	= pixels combined to improve signal to noise.

The Yx and Yy are sample spacings, in radians, and the Dx and Dy factors decorrelate the samples in horizontal and vertical directions respectively. These factors compensate for the correlation between samples which occurs when the sample spacing is closer than the intrinsic resolution of the sensor.

The Dx, Dy, Yx, and Yy terms are dependent on the specific ATR system. The POT FACTOR term defined in the Sensors block of the data set (see Section 3.3.5) is computed as the constant portion of the above equation:

$$POT\ FACTOR = \frac{Dx\ Dy}{Yx\ Yy\ N}$$

The curve relating POT to a detection probability, P_POT, is shown in Figure 2-9. The curve definition, based on empirical data, is:

$$
\begin{aligned}
P\_POT &= 0.0 && , POT < 20.0 \\
P\_POT &= 0.008\ POT + 0.16 && , 20.0 < POT < 105.0 \\
P\_POT &= 1.0 && , 105.0 < POT
\end{aligned}
$$

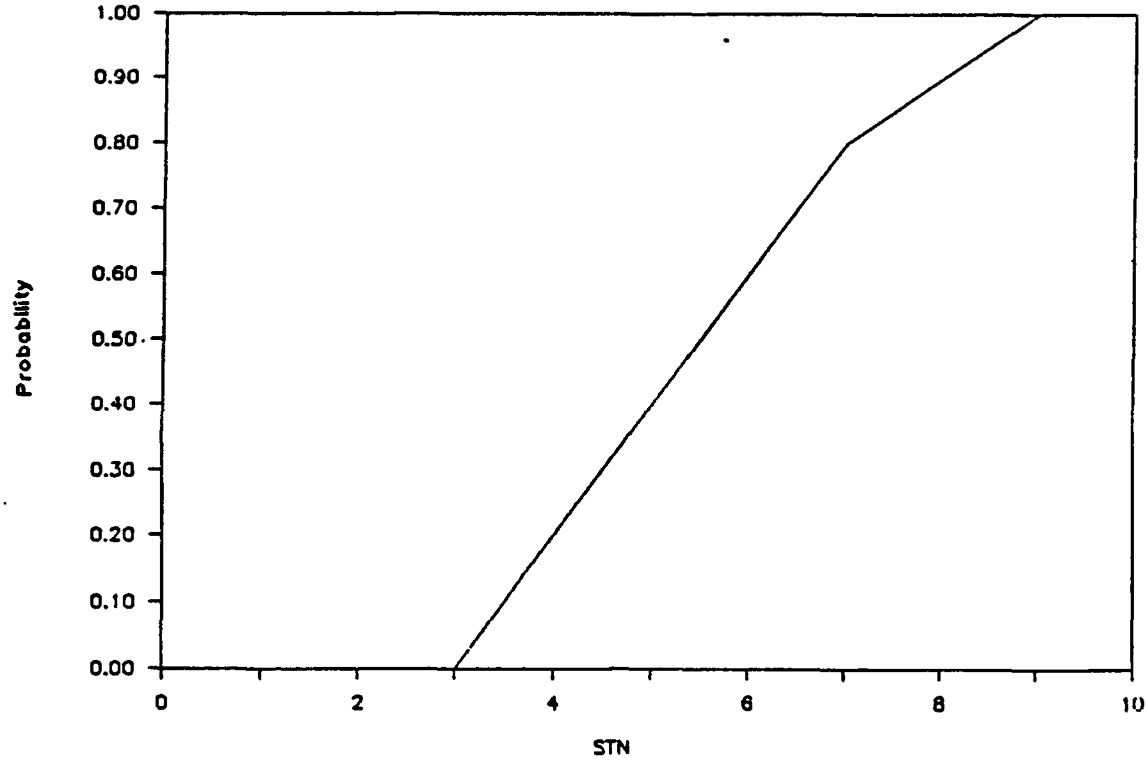

Figure 2-8, STN versus Probability of Detection

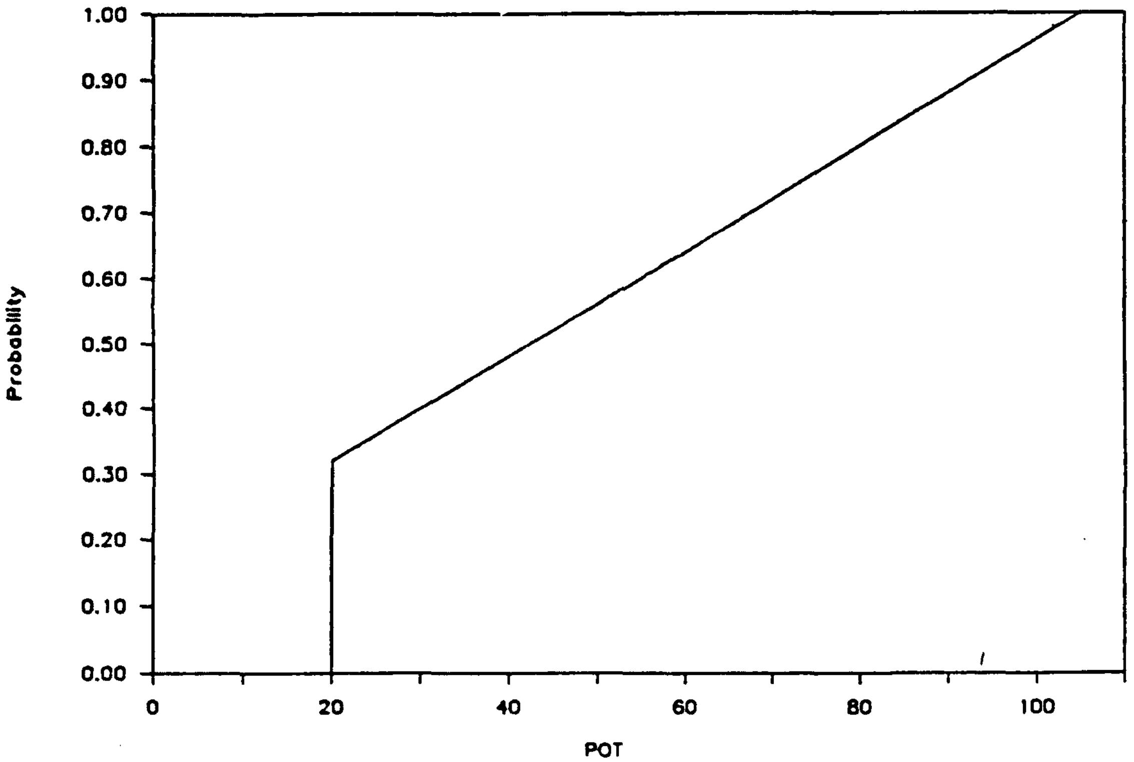

Figure 2-9, POT versus Probability of Detection

2.11.4 Probability of Detection

The probability of detection combines the previously calculated probabilities as follows:

$$P = P\_STN \;\; P\_POT$$

The model compares this probability with a sample obtained from the SIMSCRIPT pseudo-random number generator. If the sample is greater than P, no detection occurs. If the sample is less than or equal to P, the detection does occur.

2.11.5 ATR/Radar Fusion

When an ATR system is fused with a millimeter wave radar, the probability of detection becomes a fixed 90%.

2.11.6 Acquisition Level

If the detection occurred, the Johnson level at which the target was acquired must be determined. As with a regular sensor, the target's spatial frequency is interpolated from the sensor's twenty point resolution curve using the computed dT', see Section 2.10.2. An ATR system operating in automatic mode may have a different resolution curve than when it is operating in manual mode. Through the data set the user defines whether an ATR sensor has one curve for both modes or a separate curve for each.

The number of resolvable cycles is then computed, as described in Section 2.10.4. This then yields the acquisition level, as described in Section 2.10.5.

2.12 The Simulation

This section describes how the data and algorithms described in the preceding sections are combined to form the Hunter-Killer Model. For convenience, the discussion assumes that each platform has both a hunter and a killer component. A platform may have only one component (see Section 2.3.2).

2.12.1 Multiple Replications

A single replication of the model causes a set of blue and red platforms to move, acquire targets, and engage targets. During the replication a number of statistics are collected to measure the performance of the platforms in acquiring and engaging targets. Some of the data collected are the minimum, maximum, and average sensor acquisition times, the number of shots fired, the number of targets killed, the average number of targets in the field of search, and the average engagement duration. These statistics are discussed further in Sections 4.3 and 4.4 describing the model's reports.

A stochastic model should always be run for more than one replication. The model makes decisions such as whether line of sight exists or whether a shot causes a kill by comparing input probabilities with samples generated by the SIMSCRIPT II.5 pseudorandom generator. It is possible for the results of any one replication to be skewed by the particular samples generated in that replication. To avoid this problem, the Hunter-Killer Model is always run for multiple replications, typically ten to twenty. The model collects statistics for each replication and prints them at the end of the replication. The model also averages the statistics over all replications and prints summary reports.

There is no hard and fast rule for the number of replications that must be run. Past observations have shown that because the model uses only a few platforms, typically fewer than twenty, the results can vary significantly from replication to replication depending on which side makes the first acquisition and which side makes the first kill. This can be seen in the summary reports in Section 4.4. The averaged results over all replications do not, however, change significantly for most data sets after ten replications have been made. The averaged results from runs of twenty or more replications are essentially identical.

The use of the pseudorandom number generator does not imply that each run of the model using the same data set will produce different results. The pseudorandom number generator is not truly random: the generator produces a fixed sequence of samples. (SIMSCRIPT provides ten sequences of samples.) Each time the model is started the generator produces the same sequence of samples, causing two different runs made with the same data set to be identical. Each replication within a run is different, however. Replication two does not use the first sample in the sequence but instead uses the next sample after the last one used by replication one. Similarly, replication three uses the next sample in the sequence after the last one used by replication two.

2.12.2 Event Steps

There are two types of simulation models: time step and event step. In a time step model, the simulation clock is advanced a fixed amount and all activities are updated. The clock is again advanced by the same amount of time and all activities are again updated. This continues for the duration of the replication. In a time step simulation all activities always occur in the same order each time the clock is advanced.

An event step model does not advance the simulation clock by fixed amounts. Instead, when one activity occurs, a number of other activities are scheduled for the future. For example, when a platform moves, the list of targets in the platform's field of search is updated and an acquisition time for each of those targets is computed. A target acquisition event is scheduled to occur at each of those times. In addition, the time required for the platform to move to its next location is computed and that move is scheduled. The event that occurs next is determined solely by the computed times. Unlike a time step simulation, the events do not always occur in the same order.

The Hunter-Killer Model is an event step model.

2.12.3 Initialization of a Replication

At the start of each replication, each platform is initialized. This consists of the following steps:

> Reset the location to the input initial coordinates.
>
> Reset the status to alive.
>
> Reset the components' sensors to searching in wide FOV.
>
> Schedule the next move. The time the next move occurs is computed by dividing the input movement interval by the speed along the platform's first movement vector.
>
> Determine the LOS state between the platform and every other platform as discussed in Section 2.2. LOS may exist or be broken. The fraction of the target platform that is visible is also computed.
>
> For each of the platform's components, create a list of all targets that are in the component's field of search.
>
> For each of these targets, compute an acquisition time as discussed in Section 2.10. Schedule a target acquisition at this time for each component/target pair.

If the platform is airborne, additional processing is required. Much of the processing for airborne platforms is for the flight team instead of for individual platforms. In particular, movement is different than for ground platforms.

For each member of a flight team using FIXED tactics, reset the status to unmasked.

For each member of a flight team using other tactics, reset the status to masked, unless it is the scout. The scout starts unmasked.

If the team is using FIXED tactics, schedule a move as for ground platforms. The entire team moves together.

If the team is using other tactics, determine the time the scout will remask and schedule that event.

From this point on, the event that occurs next will always be the one with the shortest scheduled time. Each ground platform operates completely independently of all other platforms. Each flight team operates independently of all other platforms not in the team.

2.12.4 Ground Movement

At initialization, the next move for a ground platform is scheduled. When that move occurs, the following steps are performed:

The distance moved is computed. This is determined by multiplying the speed by the amount of time the platform has been moving.

The platform's coordinates are updated.

If the platform has moved the entire length of its current movement vector, it will switch to the next vector (if there is one). This may require the search azimuths for the hunter and killer components to be changed, as discussed in Section 2.4.2.

Acquisition times are dependent on range and, since the range to a target changes when the platform moves, the previous times are no longer valid after the move. Movement may also have caused LOS to be broken, requiring an engagement to be stopped. Consequently, all interactions with other platforms must be updated after a move:

All target acquisitions by this platform that have not yet occurred are canceled.

For each of the platform's components, update the list of all targets that are in the component's field of search.

For each of these targets, compute a new acquisition time based on the new ranges as discussed in Section 2.10. Schedule a new target acquisition at this time for the component/target pair.

If the platform was engaging a target and LOS was broken by the move, cancel the engagement.

The platform's movement also affects the actions of all other platforms. The platform may have moved into or out of the fields of search of other platforms. The range between the platform and all other platforms has changed requiring that the time required for them to acquire the first platform also be changed. As a consequence, the above steps must also be performed for all other platforms trying to acquire the one that moved.

2.12.5 Air Movement

Airborne platforms using FIXED tactics move in exactly the same manner as ground platforms, except the team members move together using the scout's movement vectors. Each movement vector will have an associated altitude at which the platform will fly while on that vector.

Airborne platforms using all other tactics only move if no targets were acquired during a popup. The team then flies masked for the input movement interval distance. During this time the team cannot acquire targets and cannot itself be acquired. The scout then unmasks to try to locate targets.

2.12.6 Target Acquisition

This section details the algorithm used by a platform to detect targets. The acquisitions are scheduled whenever a move occurs. A target in this context is any platform, either friendly or enemy. Depending on the input engagement criteria, a platform that cannot be distinguished as either friendly or enemy may be treated as an enemy and fired upon, leading to the possibility of fratricide.

Target acquisitions are always for a specific component, either the hunter or the killer, acquiring a specific target. The computations used to determine the acquisition time also provide the Johnson level at which the target is acquired. The steps in an acquisition are as follows:

> The acquisition of the target by the platform's other component is canceled. It is assumed that the acquiring component will communicate information about the target to the other component.
>
> The Johnson level at which the target is acquired is compared to the component's action level (determined through the input data set). If the acquisition level is less than the action level, the component will switch its sensor to narrow FOV.
>
> A sensor has a time delay for switching FOV during which it cannot display a focused image. The acquisition procedure must wait until this time has passed before it can continue. During this time the component cannot search for other targets.
>
> All other scheduled acquisitions by the component must be delayed by the time required to switch to narrow FOV, interrogate the target, then switch back to wide FOV to continue searching.
>
> After the delay time, the new acquisition level is determined. This is done using the equations in Section 2.10.

If the acquisition level is still less than the action level, the target is dropped and this process stops. The sensor returns to wide FOV and the component resumes searching for other targets.

If the acquisition is equal to or greater than the action level, the target is interrogated further.

The target's color at the acquisition level is determined. This is done through the input Johnson target hierarchy described in Section 2.9.

If the target is friendly -- the same color as the acquiring platform -- the target is dropped and the process stops. The sensor returns to wide FOV and the component resumes searching for other targets.

If the target is not friendly -- either belonging to the opposite side or the side cannot be distinguished -- the target will be engaged.

If the acquiring component is the killer, it begins to engage the target.

If the acquiring component is the hunter, the target is placed in the killer's acquired targets list. If the killer is not busy, it will immediately engage the target. If the killer was busy -- either interrogating another target using narrow FOV or engaging a target -- the killer will process the new target after finishing with the current one.

If the acquiring component is the hunter it returns to wide FOV and resumes searching.

The Johnson hierarchy of targets assigns a priority to a target based on the level at which it is acquired, see Section 2.9. When a killer finishes with a target it uses this priority to determine which of the targets found by the hunter will be processed next.

Flight teams using FIXED or POPUP tactics use the same procedure while unmasked. For POPUP tactics, the team will only unmask for a short period of time, determined by the data set, during which it attempts to acquire targets. Whenever the team masks, all acquisitions involving the team members are canceled.

For teams using ATR tactics, the scout unmasks only long enough to scan the field of search, then remasks. During this time it is vulnerable to being acquired and fired upon. When remasked, the scout processes the stored imagery and selects a targets. When the scout again unmasks, the target will be in the sensor's narrow FOV. See Section 2.5.3.

2.12.7 Engagements

An engagement starts when the killer acquires a target or when the hunter acquires a target and passes it to the killer. While the killer is engaging a target the hunter will continue searching for additional targets. These targets are placed in a prioritized list. After finishing an engagement the killer starts engaging the first target in this list. If there are no targets, the killer returns to searching using wide FOV.

The steps in an engagement are as follows:

All scheduled target acquisitions by the killer are canceled: the killer cannot engage and search at the same time.

The killer switches to narrow FOV if it is not already there. This requires the killer to be inactive for the switch FOV delay time.

The killer employs its laser rangefinder, it if has one. See Section 2.7.

The killer fires a single round.

If the weapon requires a target designator, the killer must wait until the round arrives at the target before it can fire another round. During this time the killer's platform may be killed itself.

A probability of kill curve is selected from the data set. The curve employed is based on the weapon used, the target's platform type, and whether rangefinding was successful. If the round required a designator and the killer's platform was killed before the round arrived at the target, the PK will be degraded. The success of the shot is assessed. See Section 2.6.

If the shot was successful, all events involving the target are canceled. If the shot was not successful, the killer waits the weapon reload time, fires another round, and repeats the assessment procedure. This continues until the target is killed.

Statistics are collected on the number of shots fired, if a kill was made, the engagement time, and other data.

If the hunter has acquired more targets while the killer was engaging, the killer selects the most important one. It repeats this procedure until no more acquired targets are available.

The killer's sensor returns to wide FOV. New target acquisitions are scheduled for all targets in the component's FOS.

An engagement may be stopped before the target has been killed. If either platform moves, LOS between them may be broken, preventing the killer from shooting. Also, the killer's platform may be killed.

When a flight team engages a target, only the scout unmasks to acquire a target. If the weapon requires a target designator such as a laser, the scout performs the designation and the attack platform never unmasks. If the weapon requires the firing platform to designate the target, such as with a wire guided missle, the attack platform must unmask. The scout remains unmasked only long enough to pass off the target. The attack platform must remain unmasked until the round arrives at the target. See Section 2.5.3.

2.12.8 Ending a Replication

The length of a replication is determined through the data set. At the time the replication ends all events that have not yet occurred are canceled. Reports are printed showing the final status of all platforms. Reports are also printed showing the performance of the sensors and platforms during the replication.

The data set also provides the number of replications to be performed. If the model has not yet completed all replications it reinitializes all platforms and performs another replication.

If all replications have been completed the model prints summary reports that average the statistics that were collected in each replication.

The reports are described in Section 4.0.

This page was intentionally left blank.

3.0 DATA PREPARATION

3.1 DATA OVERVIEW

The Hunter-Killer Model is driven by data supplied by the user which describe the performance characteristics of equipment and the overall conditions under which they operate. Very few values are coded into the model itself: to the greatest extent possible the model contains only the logic which determines how input values are used. By having the data external to the model a large variety of scenarios may be examined without requiring any changes to the model's source code. Additionally, the model does not restrict the user to an arbitrary number of data elements such as sensors or hunter-killer platforms: the number to be simulated is selected by the user in the data. Of course, the greater the quantity selected the greater the amount of memory required by the model. Processing time also increases with the number of platforms.

The data for the Hunter-Killer Model are structured in a manner to facilitate changes. Data are grouped into blocks, each block giving the data for a specific capability of the model. There is a Controls block which is concerned with the overall conditions under which a run is to be made. There are several blocks giving the performance characteristics of the equipment being modeled. Other blocks describe the hunter-killer platforms and their tactics by giving the platform types, initial locations, search tactics, and movement vectors. There is also a block describing the terrain on which the platforms will move.

The data are structured for minimal repetition. For example, each hunter-killer platform must have data describing its signature to the sensors. Since a scenario is likely to include many identical platforms - such as several occurrences of a single type of tank - giving the target signature for each platform would cause duplication of data. Instead of describing each platform's target characteristics along with its location and other platform specific data, the platform describes its type, such as type 101 which may have the name M1-TANK. The target signature for all platforms of type M1-TANK is then given only once, in the Type Platform data block. The various blocks are interrelated through the use of identification numbers assigned by the user.

Each data block is delimited by keywords and contains keywords defining the data within the block. The keywords are an integral part of the data set, allowing a user to interpret a data file easily without the need to refer to other documentation. Within each block, values given only once are preceded by descriptive keywords. Values given several times, such as the characteristics of a set of sensors, are organized into columns with each column headed by a descriptor.

The blocks into which the data have been grouped are:

Title
Controls
Sectors
Lasers
Sensors
Resolution Curves
Type Weapons
Type Platforms
SSPKs
Defilade Fractions Visible
Hunter-Killers
Movement
Air Tactics
Terrain

3.2 HUNTER-KILLER DATABASE SYSTEM

The Hunter-Killer Model is intended to be run on more than one type of mainframe computer. Currently it is being used on an IBM 4341 and a CDC Cyber 835. To avoid the necessity of maintaining completely separate databases on each mainframe being used, the Hunter-Killer Database System was developed. This system has been installed on an IBM AT computer using the dBASE III (a trademark of Ashton-Tate) database management software. The database system is not required in order to use the Hunter-Killer Model; instead, it provides a tool to simplify the maintenance of the data. The data may instead be kept directly on the mainframe on which the model is stored and executed.

The Hunter-Killer Database System is menu driven, displaying to the user a series of screens showing options from creating a new database to printing a report on a database. These screens lead the user through the system with a minimal need to refer to other documentation. The data within the database system are structured into the same blocks which appear in the run stream.

After the data have been entered into the database system, a run stream may be generated. To do so, the user selects the menu screen for run stream generation and responds to the screen's prompts. The prompts ask which mainframe is being used and which portion of the database is to be used. The system then builds the run stream in the format required by the Hunter-Killer Model. The IBM AT is then connected with the mainframe on which the model is to be run and the run stream is uploaded to that mainframe. Depending on the type of mainframe and the boards and software installed on the IBM AT, the connection to the mainframe may either be through a telephone using a modem and a communications software package or through a direct wire connection.

3.3 DATA FILE SPECIFICATIONS

This section will provide the details on constructing a data set for the Hunter-Killer Model. Each data block will be discussed separately. The purpose of the block will be given, followed by a data sample. Each value in the block will be defined.

Each data block starts with a record having a keyword naming the block. On the next record is a row of dashes used to underline the block name. The block's data follow. Each block is ended with the keyword "STOP". Within the block will be either a set of keywords identifying variables, each followed by a value, or column headings underneath of which the values appear.

The keywords must appear in the file. The input routines do not read and verify the keywords; however, the routines do use the alternation of text and numeric data to move through the file. If a data block is not required, for example of there are no lasers or if defilade fractions visible are not used, the keywords for that block must still be given.

The data do not need to appear in specific columns. The input routines read a value, skip all blanks, then read another value. Setting up the data in columns as shown in the figures in this section will make the resulting file much easier to read after it has been constructed. The rules followed for the sample data set were:

The data block name appears in column 1.

The data within the block start in column 3.

One space was left between columns.

The description of a value will be in the form:

KEYWORD MODE. DESCRIPTION: VALUES.

where

KEYWORD	is the keyword shown in the sample data set.
MODE	is the variable mode. I = integer. This may also appear as I(n) when the value must be of no more than "n" digits. D(n,m) = decimal, a total of "n" columns including the decimal point, with "m" columns to the right of the decimal point. T = text, either characters or numerals.
DESCRIPTION	is a short description of the variable.
VALUES	lists the permissible values for the variable. Text values appear in quotes to separate them from the descriptions. The quotes MUST NOT be included in the data set.

A complete data set is shown in Figure 3-1.

```
TITLE  12

  HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

  THIS RUN CONSISTS OF 4 BLUE M1 TANKS IN A STATIC POSITION PLUS A
  FLIGHT TEAM OF 2 HELICOPTERS, ONE SCOUT AND ONE ATTACK.  THE TANKS ARE
  IN DEFILADE AND ONLY SLIGHTLY VULNERABLE TO FIRE FROM THE RED SIDE.
  THE RED SIDE CONSISTS OF 5 T72 TANKS IN A CONVOY PLUS A VEHICLE
  CAPABLE OF FIRING HEATSEEKING MISSLES TO SHOOT DOWN HELICOPTERS.  THE
  CONVOY WILL PASS DIAGONALLY IN FRONT OF THE BLUE TANK POSITION.

  THIS DATA SET WAS CONSTRUCTED SOLELY TO TEST THE HUNTER-KILLER MODEL.
  ANY DATA APPEARING HERE SHOULD BE VERIFIED ELSEWHERE BEFORE BEING
  USED IN A PRODUCTION RUN.
STOP
CONTROLS

  RUN
    NUM.OF.REPS        15
    REP.LENGTH         60 MINUTES
    IN.DEFILADE        YES
    TERR.RESOLUTION    LOW
    SMOKE.SWITCH       NO
    FIXED.AZIMUTH      NO
  OUTPUT
    DATA.ECHO          YES
    REPLICATION.ECHO   YES
    REPORT.CONTROL      3
    REPORT.INTERVAL    15 MINUTES
    GENERAL.TRACE      NO
    SEARCH.TRACE       NO
    TERRAIN.TRACE      NO
  SEARCH
    VISIBILITY           1000       METERS
    MIN.DETECT.TIME         5.00    SECONDS
    LIGHT.LEVEL             0.00001 FOOT-CANDLES
    PROB.BLUE.BIFF          0.00
    PROB.RED.BIFF           0.00
    SKY.TO.GROUND.RATIOS
      BAND_.4-.7  3.00
      BAND_.4-.9  2.66
      BAND_.6-.9  2.66
    ABSORPTION.COEFFICIENTS
      BAND_3-5         0.10 1/KILOMETERS
      BAND_8-12        0.10 1/KILOMETERS
```

Figure 3-1, Sample Data Set

```
  JOHNSON.CRITERIA
       NAME     MRC  MRT

     DETECT      1.0  1.0
     CLASSIFY    2.0  2.0
     RECOGNIZE   4.0  4.0
     IDENTIFY    8.0  8.0
 PGM.DEGRADE.CURVE
   RANGE FACTOR

       0   1.00
     500   0.50
     700   0.00
STOP
SECTORS  3

  ID  HORZ FOS  VERT FOS

  11    60.00      9.00
  12   120.00     12.00
  13   100.00      9.00
STOP
LASERS  1

  ID  SPCTRM PEAK PWR XMISS DIVER APERT  NEP   PULSE FLS ALM

   1  10.6     220000  0.72  0.50 56.40  2.90 60.00     0.01
STOP
SENSORS  2

                 HORZ   VERT    FOV    SWITCH  S/N    POT
  ID  SPECTRUM   WFOV   WFOV  SCALING  DELAY  FACTOR FACTOR

   1  8-12      15.00   7.80    3.00     5.00 25.702 25.073
   2  8-12      13.50   6.75    3.00     5.50  0.000  0.000
STOP
RESOLUTION.CURVES

    1 NFOV MRT 0
      0.0000  0.0000   0.1000  1.6883   0.2000  2.7042   0.3000  3.3821
      0.4000  3.8677   0.5000  4.2319   0.6000  4.5154   0.7000  4.7422
      0.8000  4.9279   1.0000  5.2138   1.5000  5.6508   2.0000  5.8980
      3.0000  6.1678   4.0000  6.3121   5.0000  6.4020   6.0000  6.4634
      7.0000  6.5080   8.0000  6.5418   9.0000  6.5684  10.0000  6.5898
    2 NFOV MRT 0
      0.0000  0.0000   0.1000  1.6883   0.2000  2.7042   0.3000  3.3821
      0.4000  3.8677   0.5000  4.2319   0.6000  4.5154   0.7000  4.7422
      0.8000  4.9279   1.0000  5.2138   1.5000  5.6508   2.0000  5.8980
      3.0000  6.1678   4.0000  6.3121   5.0000  6.4020   6.0000  6.4634
      7.0000  6.5080   8.0000  6.5418   9.0000  6.5684  10.0000  6.5898
STOP
```

Figure 3-1, (continued)

TYPE.WEAPONS 6

ID	NAME	DESIGNATOR	SPEED	RELOAD
100	"M1 GUN"	NONE	0	5.0
110	"TOW"	FIRER	350	5.0
120	"HELLFIRE"	SCOUT	400	5.0
130	"COPPERHEAD"	SCOUT	500	5.0
200	"T72 GUN"	NONE	0	5.0
210	"HEATSEEKER"	NONE	0	5.0

STOP

TYPE.PLATFORMS 13

ID	NAME	LEVEL	OWNER
1	"TGT"	1	0
2	"TRACK"	2	1
3	"WEAPON"	2	1
5	"TANK"	3	2
6	"ATGM"	3	3
100	"M1"	4	5
101	"ARTY UNIT"	4	1
200	"T72"	4	5
201	"LAUNCHER"	4	6
700	"AIR TARGET"	1	0
800	"ATK HELI"	4	700
801	"SCOUT HELI"	4	700
802	"RPV"	4	700

ID	COLOR	PRIORITY	TYPE WPN	CLASS	HEIGHT	X-SEC	DELTA.T	REFLEC-TANCE	CONTRAST
100	BLUE	1	100	GROUND	2.4	7.9	1.50	0.50000	0.00
101	BLUE	1	130	GROUND	0.0	0.0	0.00	0.00000	0.00
200	RED	1	200	GROUND	2.3	6.5	1.50	0.00637	0.00
201	RED	1	210	GROUND	1.5	2.0	1.00	0.50000	0.00
800	BLUE	1	120	AIR	1.9	3.1	2.00	0.50000	0.00
801	BLUE	1	0	AIR	2.7	8.5	2.00	0.50000	0.00
802	BLUE	2	0	AIR	1.0	1.5	2.00	0.50000	0.00

STOP

Figure 3-1, (continued)

SSPKS

TYPE WPN	TYPE PLAT	RANGE	RANGED PK	UNRANGED PK	
100	200	0	0.90	0.80	
		500	0.86	0.61	
		1000	0.67	0.41	
		1500	0.46	0.22	
		2000	0.30	0.10	
		2500	0.19	0.05	
		2510	0.00	0.00	STOP
100	201	0	0.90	0.80	
		500	0.86	0.61	
		1000	0.67	0.41	
		1500	0.46	0.22	
		2000	0.30	0.10	
		2500	0.19	0.05	
		2510	0.00	0.00	STOP
110	200	0	0.85	0.85	
		4000	0.75	0.75	
		4020	0.00	0.00	STOP
110	201	0	0.85	0.85	
		4000	0.80	0.80	
		4020	0.00	0.00	STOP
120	200	0	0.85	0.85	
		4000	0.75	0.75	
		4020	0.00	0.00	STOP
120	201	0	0.85	0.85	
		4000	0.80	0.80	
		4020	0.00	0.00	STOP
130	200	0	0.85	0.85	
		15000	0.75	0.75	
		15020	0.00	0.00	STOP
130	201	0	0.85	0.85	
		15000	0.80	0.80	
		15020	0.00	0.00	STOP
200	100	0	0.00	0.80	
		500	0.00	0.61	
		1000	0.00	0.41	
		1500	0.00	0.22	
		2000	0.00	0.10	
		2500	0.00	0.05	
		2510	0.00	0.00	STOP
210	800	0	0.00	0.50	
		5000	0.00	0.40	
		5600	0.00	0.00	STOP
210	801	0	0.00	0.50	
		5000	0.00	0.40	
		5600	0.00	0.00	STOP

Figure 3-1, (continued)

```
 210  802        0  0.00     0.50
              5000  0.00     0.40
              5600  0.00     0.00   STOP
STOP
DEFILADE.FRACTIONS.VISIBLE
```

CLASS	TYPE PLAT	RANGE	FRACTION	
GROUND	200	0	1.00	
		6000	1.00	STOP
GROUND	201	0	1.00	
		6000	1.00	STOP
GROUND	100	0	0.30	
		3000	0.10	
		6000	0.00	STOP
AIR	200	0	1.00	
		6000	1.00	STOP
AIR	201	0	1.00	
		6000	1.00	STOP

STOP
HUNTER.KILLERS 12

ID	TYPE PLAT	COORDINATES X	COORDINATES Y	HUNTER SEN	HUNTER LAS	HUNTER SEC	HUNTER AZI	HUNTER LEV	KILLER SEN	KILLER LAS	KILLER SEC	KILLER AZI	KILLER LEV
101	100	500	250	1	0	11	90	1	1	1	11	90	3
102	100	500	275	1	0	11	90	1	1	1	11	90	3
103	100	500	300	1	0	11	90	1	1	1	11	90	3
104	100	500	350	1	0	11	90	1	1	1	11	90	3
120	800	100	800	0	0	0	0	0	1	0	11	124	1
121	800	100	800	0	0	0	0	0	1	0	11	124	1
201	200	4500	1000	0	0	0	0	0	1	0	12	264	1
202	200	4600	1010	0	0	0	0	0	1	0	12	264	1
203	200	4700	1020	0	0	0	0	0	1	0	12	264	1
204	200	4800	1030	0	0	0	0	0	1	0	12	264	1
205	200	4900	1040	0	0	0	0	0	1	0	12	264	1
206	201	5000	1050	0	0	0	0	0	1	0	12	264	1

STOP

Figure 3-1, (continued)

```
MOVEMENT

  MOVE.INTERVAL 100 METERS
  ID   SEQ AZI SPEED ALTITUDE DISTANCE

  120    1  90   10     100      3000  STOP
  201    1 264   10       0      4000
         2 270   10       0       500  STOP
  202    1 264   10       0      4100
         2 270   10       0       500  STOP
  203    1 264   10       0      4200
         2 270   10       0       500  STOP
  204    1 264   10       0      4300
         2 270   10       0       500  STOP
  205    1 264   10       0      4400
         2 270   10       0       500  STOP
  206    1 264   10       0      4500
         2 270   10       0       500  STOP
STOP
AIR.TACTICS

  POPUP.MIN.TIME.DOWN  120 SECONDS
  POPUP.MAX.TIME.UP     60 SECONDS
  ATR.MAX.TIME.UP       10 SECONDS
  ATR.TGT.RECOG.TIME     2 SECONDS
                        MOVE
  SCOUT TACTICS INTERVAL ATTACK PLATFORMS

   120  POPUP       100  121 STOP
STOP
TERRAIN.CONTROLS

  MU.LOS    1.000 MINUTES
  MU.NLOS   0.001 MINUTES
STOP
INITIAL.LINE.OF.SIGHT

  RANGE  PROB

      0  1.00
   2000  0.75
   7000  0.40
  10000  0.00
STOP
```

Figure 3-1, (continued)

3.3.1 Title

The TITLE data block provides the output reports with a description of the data. The title may consist of any number of lines. The entire title will be printed on the first page of output, regardless of whether DATA.ECHO (see Section 3.3.2, Controls) is set to "YES" or "NO". The first line of the title will be used as a heading that will appear on the top of each page of output. A sample Title data block appears in Figure 3-2.

```
TITLE  12
-----
  HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

  THIS RUN CONSISTS OF 4 BLUE M1 TANKS IN A STATIC POSITION PLUS A
  FLIGHT TEAM OF 2 HELICOPTERS, ONE SCOUT AND ONE ATTACK.  THE TANKS ARE
  IN DEFILADE AND ONLY SLIGHTLY VULNERABLE TO FIRE FROM THE RED SIDE.
  THE RED SIDE CONSISTS OF 5 T72 TANKS IN A CONVOY PLUS A VEHICLE
  CAPABLE OF FIRING HEATSEEKING MISSLES TO SHOOT DOWN HELICOPTERS.  THE
  CONVOY WILL PASS DIAGONALLY IN FRONT OF THE BLUE TANK POSITION.

  THIS DATA SET WAS CONSTRUCTED SOLELY TO TEST THE HUNTER-KILLER MODEL.
  ANY DATA APPEARING HERE SHOULD BE VERIFIED ELSEWHERE BEFORE BEING
  USED IN A PRODUCTION RUN.
STOP
```

Figure 3-2, Title Data Block

TITLE	I. The keyword starting the block must be followed by the number of lines in the title, exclusive of the dashes underlining the keyword.

The data block then consists of the title to be printed on the first page of output.

3.3.2 Controls

The CONTROLS data block defines the background conditions under which the simulation is to be run, the types of output desired, and the search parameters. A sample Controls data block appears in Figure 3-3.

```
CONTROLS
 ________
  RUN
    NUM.OF.REPS         15
    REP.LENGTH          60 MINUTES
    IN.DEFILADE         YES
    TERR.RESOLUTION     LOW
    SMOKE.SWITCH        NO
    FIXED.AZIMUTH       NO
  OUTPUT
    DATA.ECHO           YES
    REPLICATION.ECHO    YES
    REPORT.CONTROL       3
    REPORT.INTERVAL     15 MINUTES
    GENERAL.TRACE       NO
    SEARCH.TRACE        NO
    TERRAIN.TRACE       NO
  SEARCH
    VISIBILITY            1000        METERS
    MIN.DETECT.TIME          5.00     SECONDS
    LIGHT.LEVEL              0.00001  FOOT-CANDLES
    PROB.BLUE.BIFF           0.00
    PROB.RED.BIFF            0.00
    SKY.TO.GROUND.RATIOS
      BAND_.4-.7   3.00
      BAND_.4-.9   2.66
      BAND_.6-.9   2.66
    ABSORPTION.COEFFICIENTS
      BAND_3-5          0.10 1/KILOMETERS
      BAND_8-12         0.10 1/KILOMETERS
    JOHNSON.CRITERIA
        NAME      MRC  MRT
      _________  ____ ____
      DETECT      1.0  1.0
      CLASSIFY    2.0  2.0
      RECOGNIZE   4.0  4.0
      IDENTIFY    8.0  8.0
  PGM.DEGRADE.CURVE
    RANGE FACTOR
    _____ ______
        0  1.00
      500  0.50
      700  0.00
STOP
```

Figure 3-3, Controls Data Block

RUN	Keyword starting the run controls.
NUM.OF.REPS	I. The number of replications to be made. Increasing this value increases both memory usage and CPU time. To verify the format of a data set without performing any replications of the model, set NUM.OF.REPS to zero and DATA.ECHO to YES.
REP.LENGTH	I. The length of the replication, in minutes. The value must be followed by the keyword "MINUTES". Increasing this value increases CPU time.
IN.DEFILADE	T. Whether targets are in defilade: value may be "YES" or "NO". See the Defilade Fractions Visible data block.
TERR.RESOLUTION	T. Whether low or high resolution terrain is to be modeled: value may be "LOW" or "HIGH". See the Terrain data block.
SMOKE.SWITCH	T. Whether smoke is to be employed: value may be "YES" or "NO".
FIXED.AZIMUTH	T. Whether the search azimuths will change: value may be "YES" or "NO". If "YES", each platform will always use its input hunter and killer azimuths as the center of its search sectors. If the platform turns 180 degrees and moves back the way it came, the search azimuths will still be in their original directions. If "NO", the offset between the input azimuths and the platform's initial move azimuth will be computed. When the platform's move azimuth changes to a new value, the search azimuths will change to maintain the same offset. See Section 2.4.2.
OUTPUT	Keyword starting the output controls. See Section 4 for examples of all output.
DATA.ECHO	T. Whether the input data will be printed in formatted reports: value may be "YES" or "NO". This allows the user to check that the data were read correctly.
REPLICATION.ECHO	T. Whether reports are to be printed for every replication: value may be "YES" or "NO". If "NO", only summary reports will be printed at the end of the run.
REPORT.CONTROL	I. Controls when status reports are to be printed: value may be 0, 1, 2, 3, or 4. This control only functions when REPLICATION.ECHO has been set to "YES". 0 = no status reports 1 = status reports are printed at the start and the end of each replication 2 = 1 plus a report each REPORT.INTERVAL 3 = 2 plus a report each time a platform is killed 4 = 1 plus a report each time a platform is killed

REPORT.INTERVAL — I. The number of minutes between generation of status reports. The value must be followed by the keyword "MINUTES". This control only functions when REPORT.CONTROL has a value of 2 or 3.

GENERAL.TRACE — T. Whether debugging output is to be produced: value may be "YES" or "NO". Setting this value to "YES" will generate a large amount of data. This should only be used when NUM.OF.REPS is 1 and REP.LENGTH is small.

SEARCH.TRACE — T. Whether debugging output is to be produced by the search module: value may be "YES" or "NO". Setting this value to "YES" will generate a large amount of data. This should only be used when NUM.OF.REPS is 1 and REP.LENGTH is small.

TERRAIN.TRACE — T. Whether debugging output is to be produced by the line of sight module: value may be "YES" or "NO". Setting this value to "YES" will generate a large amount of data. This should only be used when NUM.OF.REPS is 1 and REP.LENGTH is small.

SEARCH — Keyword starting the search controls.

VISIBILITY — I(5). The visibility, in meters. The value must be followed by the keyword "METERS".

MIN.DETECT.TIME — D(5,2). The minimum time, in seconds, required to detect a target. The value must be followed by the keyword "SECONDS". If a smaller time is computed by the search module it will be reset to this value. See Section 2.10.7.

LIGHT.LEVEL — D(7,5). The background light level, in foot-candles. The value must be followed by the keyword "FOOT-CANDLES". This value is only used by contrast sensors.

PROB.BLUE.BIFF — D(4,2). The probability that blue platforms will successfully employ BIFF to identify enemies: value may be from 0.00 to 1.00. If BIFF is not to be modeled, set the value to 0.00.

PROB.RED.BIFF — D(4,2). The same as PROB.BLUE.BIFF, but applying to red platforms.

SKY.TO.GROUND.RATIO — Keyword starting the sky to ground ratios to determine the apparent contrast of a target.

BAND_.4-.7 — D(4,2). The sky to ground ratio used for sensors operating at .4 to .7 microns.

BAND_.4-.9 — D(4,2). The sky to ground ratio used for sensors operating at .4 to .9 microns.

BAND_.6-.9 D(4,2). The sky to ground ratio used for sensors operating at .6 to .9 microns.

ABSORBTION.COEFFICIENTS
Keyword starting the absorption coefficients. These values give the amount of infrared energy absorbed by the air.

BAND_3-5 D(4,2). The absorption level for energy in the 3 to 5 micron spectrum. The value must be followed by the keyword "1/KILOMETERS".

BAND_8-12 D(4,2). The absorption level for energy in the 8 to 12 micron range. The value must be followed by the keyword "1/KILOMETERS".

JOHNSON.CRITERIA Keyword starting the Johnson Criteria values. These values define the four acquisition levels in resolvable cycles.

NAME T. The acquisition level name: the values are "DETECT", "CLASSIFY", "RECOGNIZE", and "IDENTIFY". All four levels must be given and must appear in that order.

MRC D(4,1). The Johnson criterion for contrast sensors.

MRT D(4,1). The Johnson criterion for infrared sensors.

PGM.DEGRADE.CURVE Keywcrd starting the curve used to degrade the probability of kill by a precision guided munition when the target designator is killed before the round reaches the target.

RANGE I(5). The range, in meters, between the target and the round when the target designator is killed. The first range must be 0 (zero).

FACTOR D(4,2). The fraction of the probability of kill remaining at the associated range. At a range of zero this value should be one. The fraction should decrease as the range increases representing the decreased likelihood of hitting the target without the assistance of a designator.

3.3.3 Sectors

The SECTORS data block defines the size of the sectors in which a platform's sensors will search. A sample Sectors data block appears in Figure 3-4.

```
SECTORS  3

 ID  HORZ FOS VERT FOS

  11   60.00     9.00
  12  120.00    12.00
  13  100.00     9.00
STOP
```

Figure 3-4, Sectors Data Block

SECTORS	I. The keyword starting the data block must be followed by the number of sectors being defined.
ID	I(3). An ID number that must be unique to each sector: value may be from 1 through 999.
HORZ FOS	D(6,2). The horizontal width of the sector, in degrees.
VERT FOS	D(6,2). The vertical width of the sector, in degrees.

3.3.4 Lasers

The LASERS data block describes the operating characteristics of the laser range finders. A sample Lasers data block appears in Figure 3-5.

```
LASERS  1

   ID  SPCTRM PEAK PWR XMISS DIVER APERT  NEP  PULSE FLS ALM

    1   10.6    220000  0.72  0.50 56.40  2.90 60.00   0.01
STOP
```

Figure 3-5, Lasers Data Block

LASERS	I. The keyword starting the block must be followed by the number of lasers being modeled.
ID	I(3). An ID number that must be unique to each laser: value may be from 1 through 999.
SPCTRM	T. The wavelength of the laser: value may be "1.06" or "10.6".
PEAK PWR	I. The power level, in watts.
XMISS	D(5,2). The fraction of the signal transmitted through the optics. This value is the product fraction transmitted by the transmitter optics and fraction transmitted by the receiver optics; i.e., XMISS = T_t T_r, where T_t and T_r are defined in Section 2.7.
DIVER	D(5,2). The half angle beam divergence, in milliradians.
APERT	D(5,2). The receiver aperture, in millimeters.
NEP	D(5,2). The noise equivalent power, in nanowatts.
PULSE	D(5,2). The pulse width, in nanoseconds.
FLS ALM	D(5,2). The false alarm rate, in number per second.

3.3.5 Sensors

The SENSORS data block, along with the Resolution Curves data block, describes the performance characteristics of each sensor to be modeled. A sample Sensors data block appears in Figure 3-6.

```
SENSORS  2

                      HORZ  VERT     FOV   SWITCH  S/N    POT
   ID  SPECTRUM  WFOV  WFOV  SCALING DELAY  FACTOR FACTOR

    1  8-12      15.00  7.80   3.00    5.00 25.702 25.073
    2  8-12      13.50  6.75   3.00    5.50  0.000  0.000
STOP
```

Figure 3-6, Sensors Data Block

SENSORS	I. The keyword starting the block must be followed by the number of sensors being modeled.
ID	I(3). An ID number that must be unique to each sensor: value may be from 1 through 999.
SPECTRUM	T. The spectrum, in microns, within which the sensor operates. Value may be one of the following: ".4-.7" = visible light - eye, DVO ".4-.9" = 2nd generation image intensifiers ".6-.9" = 3rd generation image intensifiers, TVs ".7-1.1" = near infrared "3-5" = mid infrared "8-12" = far infrared
HORZ WFOV	D(5,2). The wide horizontal field of view, in degrees.
VERT WFOV	D(5,2). The wide vertical field of view, in degrees.
FOV SCALING	D(5,2). The factor used to scale the detected spatial frequency: value should be >= 1.0. See Section 2.7.2.
SWITCH DELAY	D(5,2). For a sensor with a switchable FOV, the delay in seconds after the FOV is switched before the new image is displayed by the sensor.
S/N FACTOR	D(6,3). The signal-to-noise factor for an ATR sensor. See Section 2.11.2.
POT FACTOR	D(6,3). The pixels-on-target factor for an ATR sensor. See Section 2.11.4.

The FOV of the sensor is only used to determine the time to acquire a target, which is a function of the number of fields of view in the field of search.

Sensors only search for targets in wide FOV, and so only the number of wide fields of view in the field of search is needed. Narrow FOV is used only to try to discriminate the target at a higher level or to direct the firing of a platform's weapon; consequently, an acquisition time is not required.

The resolvable spatial frequency of a target is computed using the Resolution Curves described in Section 3.3.6, which are a function of the sensor's FOV. For most sensors with a switchable FOV, only a single curve will be input which may be for either narrow or wide FOV. The spatial frequency, F, will be computed using that single curve and then scaled as follows:

$$F_{narrow} = F_{wide} * \text{FOV SCALING}$$

Since narrow FOV improves resolution, the FOV SCALING should always be greater than 1.0.

If a sensor does not have a switchable FOV, FOV SCALING should be set to 1.0 and SWITCH DELAY should be set to zero. The size of its single FOV would be given by HORZ WFOV and VERT WFOV.

3.3.6 Resolution Curves

The RESOLUTION CURVES data block, along with the Sensors data block, defines the performance characteristics of the sensors. This block gives MRC to spatial frequency curves for contrast sensors, MRT to spatial frequency curves for infrared sensors, and MDT to inverse target size curves for infrared sensors capable of MDT acquisitions. A sample Resolution Curves data block appears in Figure 3-7.

```
RESOLUTION.CURVES

    1 NFOV MRT 0
     0.0000  0.0000   0.1000  1.6883   0.2000  2.7042   0.3000  3.3821
     0.4000  3.8677   0.5000  4.2319   0.6000  4.5154   0.7000  4.7422
     0.8000  4.9279   1.0000  5.2138   1.5000  5.6508   2.0000  5.8980
     3.0000  6.1678   4.0000  6.3121   5.0000  6.4020   6.0000  6.4634
     7.0000  6.5080   8.0000  6.5418   9.0000  6.5684  10.0000  6.5898
    2 NFOV MRT 0
     0.0000  0.0000   0.1000  1.6883   0.2000  2.7042   0.3000  3.3821
     0.4000  3.8677   0.5000  4.2319   0.6000  4.5154   0.7000  4.7422
     0.8000  4.9279   1.0000  5.2138   1.5000  5.6508   2.0000  5.8980
     3.0000  6.1678   4.0000  6.3121   5.0000  6.4020   6.0000  6.4634
     7.0000  6.5080   8.0000  6.5418   9.0000  6.5684  10.0000  6.5898
STOP
```

Figure 3-7, Resolution Curves Data Block

Each sensor from the Sensors data block must have at least one curve defining its performance. Sensors may have more than one curve; for example, an infrared sensor may use one curve for narrow FOV, another for wide FOV, and a third for MDT detections.

If a sensor has a switchable FOV, as shown by its FOV SCALING being greater than one, it is not necessary to input both a narrow and wide FOV curve. Either may be input, and the computed frequency will be scaled appropriately as described in Section 3.3.5.

Sensors with a spectrum of ".4-.7", ".4-.9", or ".6-.9" are contrast sensors. The TYPE described below for a contrast sensor must be "MRC". These curves give contrast versus spatial frequency in cycles per milliradian.

For a contrast sensor, the light level at which the curve was generated must be given. If this level is not the same as for the input background level, given by control LIGHT.LEVEL, two curves must be given whose levels bracket the background level. Linear interpolation between the two curves will be used to compute the correct frequency at the background light level.

Sensors with a spectrum of ".7-1.1", "3-5", or "8-12" are infrared sensors. The TYPE described below for an infrared sensor must be set to "MRT". These curves give temperature in degrees Celsius vs spatial frequency in cycles per milliradian. An infrared sensor may also perform MDT detections. MDT curves give inverse target size in 1/milliradian vs temperature in degrees Celsius. The ability to perform MDT detections is given by the presence of an MDT curve.

Keywords are not used in this data block, except to begin and end it. Each curve is started with a record giving the following four values:

ID	I(3). The ID of a sensor. See Section 3.3.5.
FOV	T. The field of view for which the curve applies: value may be "NFOV" (narrow) or "WFOV" (wide). If the sensor has only one FOV, set the value to "WFOV". (See Section 3.3.5 for setting values for sensors with a single FOV.)
TYPE	T. The curve type: value may be "MRC", "MRT", or "MDT. For an ATR sensor operating in automatic mode the type may be "ATR".
LIGHT LEVEL	D(7,5). For MRC curves, the light level in foot-candles for which the curve applies. For MRT, MDT, or ATR curves, this value should be 0 (zero). See control LIGHT.LEVEL in Section 3.3.2.

The record with these four values must be followed by exactly twenty points that define the curve. The points must be input in the order:

X_1 Y_1 X_2 Y_2 X_3 Y_3 X_4 Y_4
...
X_{17} Y_{17} X_{18} Y_{18} X_{19} Y_{19} X_{20} Y_{20}

The order in which the curves are input does not have to be the same as the order in which the sensors were input in the previous data block.

3.3.7 Type Weapons

The TYPE WEAPONS data block describes the performance characteristics of the weapons used by the platforms. A sample Type Weapons data block is shown in Figure 3-8.

```
TYPE.WEAPONS 6

 ID      NAME       DESIGNATOR SPEED RELOAD

 100 "M1 GUN"       NONE           0    5.0
 110 "TOW"          FIRER        350    5.0
 120 "HELLFIRE"     SCOUT        400    5.0
 130 "COPPERHEAD"   SCOUT        500    5.0
 200 "T72 GUN"      NONE           0    5.0
 210 "HEATSEEKER"   NONE           0    5.0
STOP
```

Figure 3-8, Type Weapons Data Block

TYPE.WEAPONS — I. The keyword starting the block must be followed by the number of weapons being modeled.

ID — I(3). An ID number that must be unique to each type weapon: value may be from 1 through 999.

NAME — T(12). The type weapon name to be used on reports. The name may have up to ten characters. The name must be enclosed in quote marks.

DESIGNATOR — T(5). The value must be "NONE", "FIRER", or "SCOUT". This represents the platform that may server as a target designator for precision guided missles. The value "NONE" should be used for unguided rounds, such as from a tank gun, or for heat seeking missles that guide themselves. The value "FIRER" should be used for TOWs and other wire guided missles that are fired and guided by the same platform. The value "SCOUT" should be used for laser guided rounds where the target designator may be the firing platform or a completely separate platform.

SPEED — I(4). The speed of the round, in meters per second. This value is used to compute the flight time for guided rounds, where the target designator must remain exposed to fire until the round reaches the target. If the designator is killed before the round reaches the target, there will be a degradation of the PK (See PGM.DEGRADE.CURVE in Section 3.3.2). This value is ignored if the DESIGNATOR has been set to "NONE".

RELOAD — I(4,1). The time required to reload the weapon, in seconds. This time provides a throttle on the engagements to prevent an excessive number of rounds from being fired in a short period of time.

The NAME field is read using a special format that does not just read 12 characters. The first character in the name is interpreted as a delimiter. All characters between that first delimiter and the second occurrence of the same delimiter are then stored in the NAME field. This means that if, in the above example, the quote mark had not preceded the M in the name M1 GUN, the input routine would see the weapon name as being 1 GUN plus all characters up to the next M in the data file. If the final quote mark had been left off the name M1 GUN, the input routine would see the name as being every character up to the next quote mark:

```
M1 GUN    NONE          0    5.0        110
```

It would then try to read the next field, TOW, as the designator for weapon 100.

3.3.8 Type Platforms

The TYPE PLATFORMS data block defines the platform characteristics that are common to a type of platform, such as all M1 tanks. Additionally, it provides the Johnson hierarchy of targets from level 1 through level 4. See Section 2.9 for a description of this hierarchy. A sample Type Platforms data block appears in Figure 3-9.

```
TYPE.PLATFORMS  13

  ID      NAME       LEVEL OWNER

   1 "TGT"             1     0
   2 "TRACK"           2     1
   3 "WEAPON"          2     1
   5 "TANK"            3     2
   6 "ATGM"            3     3

 100 "M1"              4     5
 101 "ARTY UNIT"       4     1

 200 "T72"             4     5
 201 "LAUNCHER"        4     6

 700 "AIR TARGET"      1     0

 800 "ATK HELI"        4   700
 801 "SCOUT HELI"      4   700
 802 "RPV"             4   700
           PRIOR TYPE                                    REFLEC-
  ID COLOR  ITY  WPN  CLASS  HEIGHT X-SEC DELTA.T  TANCE  CONTRAST

 100 BLUE    1   100 GROUND   2.4    7.9   1.50   0.50000   0.00
 101 BLUE    1   130 GROUND   0.0    0.0   0.00   0.00000   0.00
 200 RED     1   200 GROUND   2.3    6.5   1.50   0.00637   0.00
 201 RED     1   210 GROUND   1.5    2.0   1.00   0.50000   0.00
 800 BLUE    1   120 AIR      1.9    3.1   2.00   0.50000   0.00
 801 BLUE    1     0 AIR      2.7    8.5   2.00   0.50000   0.00
 802 BLUE    2     0 AIR      1.0    1.5   2.00   0.50000   0.00
STOP
```

Figure 3-9, Type Platforms Data Block

The data in this block are divided in two sections. The first section defines the type platform IDs and names for all four levels along with their positions in the Johnson hierarchy. The second section defines the characteristics of the level 4 platforms.

TYPE.PLATFORMS — I. The keyword starting the block must be followed by the number of type platforms being modeled. This is the total number of type platforms over all four levels.

The first section of data is for all levels of type platforms.

ID	I(3). An ID number that must be unique to each type platform: value may be from 1 through 999.
NAME	T(12). The type platform name to be used on reports. The name may have up to ten characters. The name must be enclosed in quote marks. See Section 3.3.7 for details on how a name field is read.
LEVEL	I(1). The Johnson level of the platform: value may be 1, 2, 3, or 4.
OWNER	I(3). The ID of the owning type platform in the Johnson hierarchy. A level 1 platform, which has no owner, should be given an owner of 0 (zero).

The second section of data is for level 4 platforms only.

ID	I(3). The ID of a level 4 type platform. This must be one of the IDs given in the first section of type platform data.
COLOR	T. The side using the type platform: value may be "BLUE" or "RED".
PRIORITY	I(2). The priority of this type of platform when in a list of targets waiting to be fired on. Low numbers indicate a high priority.
TYPE WPN	I(3). The ID of the platform's type weapon. See Section 3.3.7. If the platform does not have a weapon, enter a 0 (zero).
CLASS	T. The platform's class: value may be "GROUND" or "AIR".
HEIGHT	D(4,1). The height of the platform, in meters. This value should give the target dimension which is most critical to its being detected, which is usually the height.
X-SEC	D(4,1). The platform's cross-sectional area, in square meters. This is used by laser rangefinders in computing the returned signal.
DELTA.T	D(5,2). The difference, in degrees Celsius, between the platform's temperature, measured at the platform, and the background temperature. This is used by infrared sensors.
REFLECTANCE	D(7,5). The amount of a signal reflected by the platform. This is used by laser rangefinders.
CONTRAST	D(4,2). The contrast of the platform. This is used by contrast sensors.

3.3.9 SSPKs

The SSPKs data block describes the probability of each of the type weapons killing each of the level 4 type platforms as a function of range. A sample SSPKs data block appears in Figure 3-10.

SSPKS

TYPE WPN	TYPE PLAT	RANGE	RANGED PK	UNRANGED PK	
100	200	0	0.90	0.80	
		500	0.86	0.61	
		1000	0.67	0.41	
		1500	0.46	0.22	
		2000	0.30	0.10	
		2500	0.19	0.05	
		2510	0.00	0.00	STOP
100	201	0	0.90	0.80	
		500	0.86	0.61	
		1000	0.67	0.41	
		1500	0.46	0.22	
		2000	0.30	0.10	
		2500	0.19	0.05	
		2510	0.00	0.00	STOP
110	200	0	0.85	0.85	
		4000	0.75	0.75	
		4020	0.00	0.00	STOP
110	201	0	0.85	0.85	
		4000	0.80	0.80	
		4020	0.00	0.00	STOP
120	200	0	0.85	0.85	
		4000	0.75	0.75	
		4020	0.00	0.00	STOP
120	201	0	0.85	0.85	
		4000	0.80	0.80	
		4020	0.00	0.00	STOP
130	200	0	0.85	0.85	
		15000	0.75	0.75	
		15020	0.00	0.00	STOP
130	201	0	0.85	0.85	
		15000	0.80	0.80	
		15020	0.00	0.00	STOP
200	100	0	0.00	0.80	
		500	0.00	0.61	
		1000	0.00	0.41	
		1500	0.00	0.22	
		2000	0.00	0.10	
		2500	0.00	0.05	
		2510	0.00	0.00	STOP

Figure 3-10, SSPKs Data Block

210	800	0	0.00	0.50	
		5000	0.00	0.40	
		5600	0.00	0.00	STOP
210	801	0	0.00	0.50	
		5000	0.00	0.40	
		5600	0.00	0.00	STOP
210	802	0	0.00	0.50	
		5000	0.00	0.40	
		5600	0.00	0.00	STOP
STOP					

Figure 3-10, (continued)

TYPE WPN — I(3). The ID of a type weapon. See Section 3.3.7.

TYPE PLAT — I(3). The ID of a level 4 type platform. See Section 3.3.8.

RANGE — I(5). The range, in meters, between the weapon and the targeted platform. The first range for a weapon/target combination must be 0 (zero).

RANGED PK — D(4,2). The probability of a single round hitting and killing the target at the associated range: value must be from 0.00 through 1.00. This value is used when the killer component of the firing platform has a laser rangefinder and the rangefinding was successful.

UNRANGED PK — D(4,2). The probability of a single round hitting and killing the target at the associated range: value must be from 0.00 through 1.00. This value is used when the killer component of the firing platform does not have a laser rangefinder or has a laser rangefinder and the rangefinding was not successful.

Only the first record of the SSPK curve for a weapon/target combination should give the IDs. The remaining records should begin with RANGE. The set of values for each combination must end with the keyword "STOP".

Linear interpolation is used to determine the SSPK for a range that is not one of the points given.

If a weapon/target combination is not given, that weapon cannot kill that type of platform.

3.3.10 Defilade Fractions Visible

The DEFILADE FRACTIONS VISIBLE data block defines the fraction of each type platform that will be visible as a function of range and observer class. The observer classes are AIR or GROUND. These fractions are used to model micro-terrain, as described in Section 2.2.4. These data are only used when the control IN.DEFILADE is set to "YES", see Section 3.3.2. A sample Defilade Fractions Visible data block appears in Figure 3-11.

```
DEFILADE.FRACTIONS.VISIBLE
______________________________
           TYPE
  CLASS    PLAT   RANGE FRACTION
  ______   ____   _____ ________
  GROUND   200        0   1.00
                   6000   1.00  STOP
  GROUND   201        0   1.00
                   6000   1.00  STOP
  GROUND   100        0   0.30
                   3000   0.10
                   6000   0.00  STOP
  AIR      200        0   1.00
                   6000   1.00  STOP
  AIR      201        0   1.00
                   6000   1.00  STOP
STOP
```

Figure 3-11, Defilade Fractions Data Block

CLASS	T. The observer class: value may be "GROUND" or "AIR".
TYPE PLAT	I(3). The ID of a targeted type platform. See Section 3.3.8.
RANGE	I(5). The range, in meters, between the observer and the target. The first range for a class/target combination must be 0 (zero).
FRACTION	D(4,2). The fraction of the target visible at the associated range.

Only the first record of the fraction visible curve for a class/target combination should give the CLASS and TYPE PLAT. The remaining records for the combination should begin with RANGE. The set of values for each combination must end with the keyword "STOP".

Linear interpolation is used to determine the fraction visible for a range that is not one of the points given.

A fraction visible curve should not be given for airborne targets. With low resolution terrain, an unmasked airborne platform is always 100 percent visible to both ground and air observers. With high resolution terrain, the fraction of an unmasked platform that is visible is determined solely from the input terrain elevations. A masked airborne platform is 0 percent visible to both ground and air observers.

If a class/target combination is not given, that class of observer cannot detect that type of platform.

If micro-terrain is not being modeled, as indicated by the control IN.DEFILADE being set to "NO", the keyword starting the data block, the column headings, and the keyword "STOP" at the end of the block must still appear. If two comparison runs are being made, one with IN.DEFILADE set to "YES" and the other with it set to "NO", the fractions do not need to be deleted from the "NO" data set. The model will read the values but not use them.

Note that in the sample data set, the IN.DEFILADE control is set to "YES". The curves for observers looking at red platforms are a constant 1.0. This causes the red platforms to always be 100% visible as they move. The curves for observers looking at blue platforms cause only a small fraction of a blue platform to be visible. This represents the static blue platforms being in defilade in a defensive position as they wait for targets.

3.3.11 Hunter-Killers

The HUNTER-KILLERS data block describes the individual platforms included in the simulation. Only data specific to a platform are given here. Data common to a type of platform are given in the Type Platforms data block. A sample Hunter-Killers data block appears in Figure 3-12.

HUNTER.KILLERS 12

ID	TYPE PLAT	COORDINATES X	COORDINATES Y	HUNTER SEN	HUNTER LAS	HUNTER SEC	HUNTER AZI	HUNTER LEV	KILLER SEN	KILLER LAS	KILLER SEC	KILLER AZI	KILLER LEV
101	100	500	250	1	0	11	90	1	1	1	11	90	3
102	100	500	275	1	0	11	90	1	1	1	11	90	3
103	100	500	300	1	0	11	90	1	1	1	11	90	3
104	100	500	350	1	0	11	90	1	1	1	11	90	3
120	800	100	800	0	0	0	0	0	1	0	11	124	1
121	800	100	800	0	0	0	0	0	1	0	11	124	1
201	200	4500	1000	0	0	0	0	0	1	0	12	264	1
202	200	4600	1010	0	0	0	0	0	1	0	12	264	1
203	200	4700	1020	0	0	0	0	0	1	0	12	264	1
204	200	4800	1030	0	0	0	0	0	1	0	12	264	1
205	200	4900	1040	0	0	0	0	0	1	0	12	264	1
206	201	5000	1050	0	0	0	0	0	1	0	12	264	1

STOP

Figure 3-12, Hunter-Killers Data Block

HUNTER.KILLERS — I. The keyword starting the block must be followed by the number of platforms being modeled.

ID — I(3). An ID number that must be unique to each platform: value may be from 1 through 999.

TYPE PLAT — I(3). The ID of the level 4 type platform that describes the platform. See Section 3.3.8.

COORDINATES — I(5). The X and Y coordinates, in meters, of the platforms's position at the start of each replication.

The following data are given first for the hunter component then for the killer component. If the platform does not have a hunter component, set the values for that component to 0 (zero). All five values must be given for each component.

SEN — I(3). The ID of the component's sensor. See Section 3.3.5.

LAS — I(3). The ID of the component's laser range finder. This value is ignored for the hunter. See Section 3.3.4.

SEC	I(3). The ID of the component's search sector. See Section 3.3.3.
AZI	I(3). The azimuth that defines the center of the component's search sector, in degrees: value may be from 0 through 360. The value is measured clockwise from 0 degrees = North. See control FIXED.AZIMUTH in Section 3.3.2.
LEV	I(1). The Johnson level at which the component must acquire a target before taking action: value may be one of the following 1 = detect 2 = classify 3 = recognize 4 = identify

If a platform is to have only one component, that component must be the killer. If it is desired to have a platform acquire targets but not shoot, such as with an RPV or a scout helicopter, the platform's TYPE PLAT should not have a weapon.

All of the blue platforms should always be given before the red platforms are given. The report routines assume that the platforms are given in this order. A platform's color is defined by the color of its TYPE PLAT.

3.3.12 Movement

The MOVEMENT data block defines how the platforms in the Hunter-Killers data block will move. If a ground platform does not have a move vector included in this block it will remain in its initial position. Only the scout in a flight team will have a move vector set. All other airborne team members will move with the leader using those vectors. A sample Movement data block appears in Figure 3-13.

```
MOVEMENT

  MOVE.INTERVAL 100 METERS
  ID  SEQ AZI SPEED ALTITUDE DISTANCE

  120   1  90  10      100      3000  STOP
  201   1 264  10        0      4000
        2 270  10        0       500  STOP
  202   1 264  10        0      4100
        2 270  10        0       500  STOP
  203   1 264  10        0      4200
        2 270  10        0       500  STOP
  204   1 264  10        0      4300
        2 270  10        0       500  STOP
  205   1 264  10        0      4400
        2 270  10        0       500  STOP
  206   1 264  10        0      4500
        2 270  10        0       500  STOP
STOP
```

Figure 3-13, Movement Data Block

The first value given applies to all ground platforms:

MOVE.INTERVAL — I(3). The distance, in meters, a ground platform moves each time its position is updated. The value must be followed by the keyword "METERS".

This value is followed by the move vectors for all platforms. The vectors do not have to be given in the same order as the hunter-killers were given in the previous data block.

ID — I(3). The ID of one of the platforms. See Section 3.3.11.

SEQ — I(2). The sequence number of the vector. The vectors for each platform should be numbered from 1 to n.

AZI — I(3). The direction of movement, in degrees: value may be from 0 through 360. The value is measured clockwise from 0 = North.

SPEED — I(3). The speed, in kilometers per hour.

ALTITUDE	I(5). For airborne platforms only, the altitude, in meters, at which it flies along this vector. For ground platforms this value should be 0 (zero).
DISTANCE	I(5). The length of this vector, in meters.

Only the first vector for a platform should have the ID. The second and later vectors should start with SEQ. The list of vectors for a platform must end with the keyword "STOP".

A closed path may be defined for a platform. This may be used for airborne platforms such as RPVs which are flying a closed loop. To define a closed path, list the vectors in the path. After the last vector, enter:

GOTO nn

where "nn" is the SEQ for the first vector in the loop. This does not need to be the first vector in the set: the platform may fly a straight path to its operating zone, then start flying a closed loop. The platform will move to the last vector in the set, then use the AZI, SPEED, ALTITUDE, and DISTANCE of vector "nn", then cycle to the end of the set again. The "STOP" for the platform's vector set must follow the "GOTO nn".

3.3.13 Air Tactics

The AIR TACTICS data block describes the manner in which airborne platforms will function. A sample Air Tactics data block appears in Figure 3-14.

```
AIR.TACTICS

    POPUP.MIN.TIME.DOWN  120 SECONDS
    POPUP.MAX.TIME.UP     60 SECONDS
    ATR.MAX.TIME.UP       10 SECONDS
    ATR.TGT.RECOG.TIME     2 SECONDS
                       MOVE
    SCOUT TACTICS INTERVAL ATTACK PLATFORMS
    ----- ------- -------- ----------------
     120  POPUP      100   121 STOP
STOP
```

Figure 3-14, Air Tactics Data Block

The first four values are used by all flight teams.

POPUP.MIN.TIME.DOWN
: I. The amount of time, in seconds, that a flight team using popup tactics will mask after firing a round at a target. After this time the scout will unmask at the same position to determine if there are more targets available. The value must be followed by the keyword "SECONDS".

POPUP.MAX.TIME.UP
: I. The maximum time, in seconds, that a scout in a flight team using popup tactics will unmask to attempt to acquire targets. If a target is not acquired within this time, the team will mask and move to a new position. The value must be followed by the keyword "SECONDS".

ATR.MAX.TIME.UP
: I. The amount of time, in seconds, that a scout with an ATR sensor will unmask to allow the sensor to scan the field of search. The scout will then mask to allow the sensor to process the collected data. The value must be followed by the keyword "SECONDS".

ATR.TGT.RECOG.TIME
: I. The amount of time, in seconds, required by the operator of an ATR sensor to process each target that has been acquired. The value must be followed by the keyword "SECONDS".

The remaining values describe the flight teams that have been included in the simulation.

SCOUT
: I(3). The ID number of the airborne platform serving as the team's scout.

TACTICS	T. The tactics used by the team. The value may be "FIXED", "POPUP", "ATR", or "ATRMMW".
MOVE INTERVAL	I(3). The distance, in meters, the team moves each time its position is updated.
ATTACK PLATFORMS	I(3). The ID numbers of the platforms, airborne or ground, that will fire on the targets acquired by the scout. The scout should not be included in this list even if it has weapons and will also act as an attack platform. The list must be ended with the keyword "STOP".

If no flight teams are included in the data set the keywords must still appear. The first four records must have dummy values to allow the read routine to move through the keywords correctly.

3.3.14 Terrain

The TERRAIN data block defines the terrain on which the platforms will move. As described in Section 2.2, terrain may be low or high resolution. Both data formats are defined in this section. Sample Terrain data blocks appear in Figures 3-15 and 3-16.

3.3.14.1 Low Resolution Terrain. Low resolution terrain is defined by a small data block that is included with the other data in a single data file. The block consists of two sections. Low resolution terrain is selected by setting the control TERR.RESOLUTION to "LOW", see Section 3.3.2. A sample low resolution Terrain data block appears in Figure 3-15.

```
TERRAIN.CONTROLS
____________________
  MU.LOS    1.000 MINUTES
  MU.NLOS   0.001 MINUTES
STOP
INITIAL.LINE.OF.SIGHT
____________________
  RANGE  PROB
  _____  ____
      0  1.00
   2000  0.75
   7000  0.40
  10000  0.00
STOP
```

Figure 3-15, Low Resolution Terrain Data Block

The first section of data is headed "TERRAIN.CONTROLS" and gives the mean times that line of sight between pairs of ground platforms will exist or be broken.

MU.LOS — D(6,3). The mean time, in seconds, that line of sight will exist between pairs of ground platforms. The value must be followed by the keyword "MINUTES".

MU.NLOS — D(6,3). The mean time, in seconds, that line of sight will NOT exist between pairs of ground platforms. The value must be followed by the keyword "MINUTES".

A second section of data headed "INITIAL.LINE.OF.SIGHT" follows the mean times. This section gives a curve that defines the probability of line of sight as a function of range. This curve is used at the start of each replication to determine whether line of sight exists between each pair of ground platforms.

RANGE — I(5). The range between two platforms, in meters. The first range must be 0.

PROB D(4,2). The probability that line of sight will exist at the associated range: value must be between 0.00 and 1.00.

Linear interpolation is used to determine the initial probability of line of sight for a range that is not one of the points given.

3.3.14.2 High Resolution Terrain. High resolution terrain is defined by a large data block which gives elevations, in meters, at fixed intervals. High resolution terrain is selected by setting the control TERR.RESOLUTION to "HIGH", see Section 3.3.2. A sample high resolution Terrain data block appears in Figure 3-16.

On an average sized battlefield used by this model, 4000 by 1000 meters, with elevations given at approximately 100 meter intervals, a total of 451 values must be given. Larger battlefields or smaller intervals will increase the number of values. If the data are taken from a Defense Mapping Agency Level 1 Terrain tape, the intervals will range from 70 to over 100 meters depending on the portion of the world being described. It is assumed that a set of battlefield descriptions will be extracted from the DMA tapes and then be used without further changes. For this reason, the high resolution terrain data block is contained in a separate file from the remainder of the data.

Extraction of data from a DMA data tape is described in the Programmer's Manual.

```
TERRAIN.CONTROLS
----------------
  LATITUDE              311000N
  LONGITUDE            1105000W
  BATTLEFIELD.LENGTH    4049.4 METERS
  FLOT.WIDTH            2037.2 METERS
  LOS.X.INTERVAL          79.4 METERS
  LOS.Y.INTERVAL          92.6 METERS
STOP
ELEVATIONS
----------

     1798   1828   1853   1871   1898   1897   1896   1898   1879   1860
     1842   1822   1804   1795   1786   1764   1736   1699   1688   1671
     1649   1626   1600

     ........

     1782   1799   1826   1846   1875   1898   1897   1896   1895   1877
     1861   1841   1818   1796   1793   1773   1741   1710   1691   1677
     1659   1637   1616
STOP
```

Figure 3-16, High Resolution Terrain Data Block

The first section of data, headed "TERRAIN.CONTROLS", defines the size of the battlefield.

LATITUDE — T. The latitude of the lower lefthand corner of the battlefield.

LONGITUDE — T. The longitude of the lower lefthand corner of the battlefield.

BATTLEFIELD.LENGTH D(7,1). The length, in meters, of the battlefield measured along the X-axis. This should be an integer multiple of LOS.X.INTERVAL. The value must be followed by the keyword "METERS".

FLOT.WIDTH D(7,1). The width, in meters, of the battlefield measured along the Y-axis. This should be an integer multiple of LOS.Y.INTERVAL. The value must be followed by the keyword "METERS".

LOS.X.INTERVAL D(5,1). The interval along the X-axis, in meters, between elevations. The value must be followed by the keyword "METERS".

LOS.Y.INTERVAL D(5,1). The interval along the Y-axis, in meters, between elevations. The value must be followed by the keyword "METERS".

The LATITUDE and LONGITUDE values are included in the data file solely to indicate the source of the data. The model does not use the values, but does read and echo them. The values must be given in the form "dddmmssh", where "ddd" is degrees, "mm" is minutes, "ss" is seconds, and "h" is the hemisphere ("N", "S", "E", or "W"). Each value must be given as one contiguous field with no embedded blanks. Minutes and seconds less than 10 must be given with a leading zero; for example, 5 minutes must be given as 05 minutes.

The second section of data, headed "ELEVATIONS", gives the elevations in integer meters. Assume the coordinates of the battlefield run from X_0 to X_n along the X-axis and from Y_0 to Y_m along the Y-axis. The spacing between coordinates is given by LOS.X.INTERVAL and LOS.Y.INTERVAL. Then the elevations are read for locations in the order:

X_0Y_0 X_0Y_1 X_0Y_2 X_0Y_3 X_0Y_4 ...... X_0Y_m
X_1Y_0 X_1Y_1 X_1Y_2 X_1Y_3 X_1Y_4 ...... X_1Y_m
......
X_nY_0 X_nY_1 X_nY_2 X_nY_3 X_nY_4 ...... X_nY_m

The number of values along a constant Y-coordinate will be:

1 + BATTLEFIELD.LENGTH / LOS.X.INTERVAL

The number of values along a constant X-coordinate will be:

1 + FLOT.WIDTH / LOS.Y.INTERVAL

The values along a constant X-coordinate, from X_0Y_0 to X_0Y_m for example, may extend across several records. The sample data set has ten values per record, allowing 5 columns per elevation with two spaces between each value. For improved readability, a blank record separates the set of values for each constant X-coordinate.

3.4 INPUT ERRORS

There are two classes of errors that may be detected when data are being read. The first class is SIMSCRIPT detected errors. These will occur when the data have been incorrectly formatted so that an expected data type is not found. For example, the input routines may try to skip a keyword, then read a numeric variable. If the keyword has been omitted from the file, the field that the routines will skip will be the variable, not the keyword. Then instead of the next field being the variable it may be the next keyword. SIMSCRIPT will detect a text field instead of a number and will generate an error message and stop. Also, if a numeric value with a decimal point is entered where an integer is expected, the incorrect mode will be detected.

The second class of errors consists of those detected by the Hunter-Killer Model itself. These occur when an input value is checked to make sure it is within the permissible range. For example, if the color of a type target was entered as "BLEU", the model would detect that this is not either of the valid values of "BLUE" or "RED". An error message would be printed and the model would stop. The other most likely error that may occur is that an ID in one data block does not cross-reference with the IDs in another data block correctly. For example, if the Movement data block has vectors for a hunter-killer with an ID of 300 and if the Hunter-Killer data block did not have a platform with that ID, an error message would be printed and the model would stop.

When an error is detected by the model, an error message, the record being read, and a traceback will be printed. The information that is displayed is as follows:

```
READ.r: message

ERROR READING RECORD = nnn, COLUMN = mmm

THE RECORD BEING READ WAS:
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

traceback
```

where READ.r is the name of the routine reading the data
- message is the error message
- nnn is the number of the record being read
- mmm is the column that ends the field being read
- xxxx is the record

The relationship between the input routines and the data blocks is:

READ.A	Air Tactics
READ.C	Controls
READ.D	Title, Sectors
READ.F	Defilade Fractions Visible
READ.G	Type Platforms
READ.H	Hunter-Killers
READ.L	Lasers
READ.M	Movement

READ.P SSPKs
READ.R Resolution Curves
READ.S Sensors
READ.T Terrain
READ.W Type Weapons

When SIMSCRIPT detects the error, the information given differs slightly. The order will be:

SIMSCRIPT error message

traceback

ERROR READING RECORD = nnn, COLUMN = mmm

THE RECORD BEING READ WAS:
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Since the SIMSCRIPT error messages are different for each type of computer, they will not be shown here. Refer to the appropriate User's Manuals listed in Appendix A.

It should usually be unnecessary to examine the traceback to determine the line of code being executed when the error occurred. Examining the record being read, the error message, and the format of the data set should be sufficient.

When a new data set is constructed it may be desirable to verify that the data are in the correct format before continuing with the simulation. To do this, set the control NUM.OF.REPS to 0 and set DATA.ECHO to YES. The model will then read the data, produce the data echo reports, then stop. The user may then verify that the data were read correctly. If they were read correctly, reset NUM.OF.REPS to the desired number of replications and restart the job.

3.5 CPU TIME AND MEMORY CONSIDERATIONS

There are several factors that can greatly affect the amount of CPU time and memory required to perform a simulation. Time is most greatly affected by the controls NUM.OF.REPS, REP.LENGTH, and TERR.RESOLUTION; by the number of hunter-killers; and by the size of the ground and air move intervals. The controls NUM.OF.REPS and REP.LENGTH directly increase the processing time by increasing the number and duration of replications. If TERR.RESOLUTION is set to "HIGH", the amount of processing increases greatly since the line of sight computations are no longer a simple yes or no test, see Section 2.2. The number of hunter-killer platforms determines how many sensors are being used and how many observer/target pairs there are between which line of sight must be determined and time to acquire must be calculated. While neither line of sight nor time to acquire is a time consuming calculation, both are performed a very large number of times. The time to acquire a target is a function of range and must be recomputed each time range changes. As the move intervals decrease in size, the number of times the time to acquire must be recomputed increases. Naturally, the line of sight calculations must also be made each time a platform moves.

Memory is most greatly affected by the control NUM.OF.REPS and, if high resolution terrain is used, the size of the battlefield. The data used to produce the histograms that are generated by the model (see Section 4.4) are stored in four three-dimensional arrays, one dimension of which is the number of replications. As this value grows, the amount of memory required to store these arrays increases quickly. When high resolution terrain is used, the elevations are stored in a two-dimensional array. As the size of the battlefield grows or the interval between elevations decreases, the number of elevations to be stored, and consequently the size of this array, increases.

This page was intentionally left blank.

4.0 REPORT DESCRIPTIONS

4.1 REPORT CLASSES

The Hunter-Killer Model provides the user with detailed output of the simulation results to support analysis. The generation of all reports except the final summaries, which are always generated, is at the option of the user. The report classes are: Data Echo, Replication, Final Summaries, and Debugging. Examples of all of these reports are shown in the figures in this section.

The reports appear in two output files. Through the run stream the user must save or print these files. The names of these files are determined by the user when building a run procedure. Internal to the model the files are referred to by the logical unit numbers 6 and 8.

Unit 6 will contain, at a minimum, progress messages printed before reading the data, after echoing the data, and before and after each replication. The presence of these messages confirms that the run was executed as intended. Additionally, the Debugging Reports, if selected, will appear in this file. If there is an error which causes the run to abort, the traceback generated by SIMSCRIPT II.5 will appear in this file. (Note: On some systems, SIMSCRIPT writes the error traceback to a separate file.)

Unit 8 will contain the majority of the output.

As seen in the following figures, every page of output starts with a two line header. The first line gives the version of the Hunter-Killer Model being executed - Version 2.0 - along with the date the run was made and the time at which the run was started. The first line also states at what portion of the simulation the report was printed: Model Initialization, Replication nn (where "nn" is the number of the replication), or Final Reports. The second line of the header is the first line of the title that was read from the run stream (see Section 3.3.1).

4.2 DATA ECHO REPORTS

The Data Echo Reports which appear on unit 8 are generated when the variable DATA.ECHO is set to "YES". If DATA.ECHO is set to "NO", the reports will not be generated. In either case, the run description read from the TITLE data block will be printed on the first page of output. It is recommended that these reports always be generated to avoid later confusion about which data were used to generate the remaining reports. These reports should be carefully examined to ensure that all data were read correctly.

Since these reports simply echo the input data, they will not be described in detail. The reports generated by the sample run stream from Section 3 appear in Figure 4-1.

```
=======================================================================
=======================================================================

                        TTTTT  H  H  EEEE
                          T    H  H  E
                          T    HHHH  EEE
                          T    H  H  E
                          T    H  H  EEEE

H  H  U  U  N  N  TTTTT  EEEE  RRR     K  K  I  L     L     EEEE  RRR
H  H  U  U  NN N    T    E     R  R    K K   I  L     L     E     R  R
HHHH  U  U  N NN    T    EEE   RRR  -- KK    I  L     L     EEE   RRR
H  H  U  U  N  N    T    E     R R     K K   I  L     L     E     R R
H  H   UU   N  N    T    EEEE  R  R    K  K  I  LLLL  LLLL  EEEE  R  R

                  M   M   OO   DDD   EEEE  L
                  MM MM  O  O  D  D  E     L
                  M M M  O  O  D  D  EEE   L
                  M   M  O  O  D  D  E     L
                  M   M   OO   DDD   EEEE  LLLL

=======================================================================
=======================================================================
```

RUN STARTED ON 16-SEP-1986 AT 14:57

R U N   D E S C R I P T I O N

HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

THIS RUN CONSISTS OF 4 BLUE M1 TANKS IN A STATIC POSITION PLUS A FLIGHT TEAM OF 2 HELICOPTERS, ONE SCOUT AND ONE ATTACK. THE TANKS ARE IN DEFILADE AND ONLY SLIGHTLY VULNERABLE TO FIRE FROM THE RED SIDE. THE RED SIDE CONSISTS OF 5 T72 TANKS IN A CONVOY PLUS A VEHICLE CAPABLE OF FIRING HEATSEEKING MISSLES TO SHOOT DOWN HELICOPTERS. THE CONVOY WILL PASS DIAGONALLY IN FRONT OF THE BLUE TANK POSITION.

THIS DATA SET WAS CONSTRUCTED SOLELY TO TEST THE HUNTER-KILLER MODEL. ANY DATA APPEARING HERE SHOULD BE VERIFIED ELSEWHERE BEFORE BEING USED IN A PRODUCTION RUN.

Figure 4-1, Data Echo Report

```
HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57      : MODEL INITIALIZATION
  HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

                              C O N T R O L S

RUN                                          OUTPUT
---                                          ------
  NUM OF REPLICATIONS     15                   DATA ECHO             YES
  REPLICATION LENGTH      60 MINUTES           REPLICATION ECHO      YES
  TARGETS IN DEFILADE     YES                  REPORT CONTROL          3
  TERRAIN RESOLUTION      LOW                  REPORT INTERVAL        15 MINUTES
  SMOKE                   NO                   GENERAL TRACE         NO
  FIXED AZIMUTH           NO                   SEARCH  TRACE         NO
                                               TERRAIN TRACE         NO

SEARCH
------
  VISIBILITY              1000        METERS
  MIN DETECT TIME            5.00     SECONDS
  LIGHT LEVEL                 .00001  FOOT-CANDLES
  PROB BLUE BIFF             0.
  PROB RED  BIFF             0.
```

SPECTRA

BAND	SKY TO GROUND RATIO	ABSORPTION COEFFICIENT	EXTINCTION COEFFICIENT
.4-.7	3.000	0.	3.9120
.4-.9	2.660	0.	3.6222
.6-.9	2.660	0.	2.8495
.7-1.1	0.	0.	3.5252
3-5	0.	.100	1.7571
8-12	0.	.100	1.2419
1.06	0.	0.	3.5252
10.6	0.	.100	1.2419

JOHNSON CRITERIA

NAME	N50 CYCLES MRC	MRT
DETECT	1.0	1.0
CLASSIFY	2.0	2.0
RECOGNIZE	4.0	4.0
IDENTIFY	8.0	8.0

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

C O N T R O L S (PART 2)

PGM DEGRADE CURVE

RANGE (METERS)	DEGRADE FACTOR
0	1.00
500	.50
700	0.

S E C T O R S

SECTOR ID	HFOS (DEGREES)	VFOS (DEGREES)
11	60.00	9.00
12	120.00	12.00
13	100.00	9.00

L A S E R S

ID	SPECTRUM	PEAK POWER (WATTS)	OPTICS XMISS	BEAM DIVERG (MRAD)	APER-TURE (MM)	NOISE EQUIVALENT POWER (NANOWTTS)	PULSE WIDTH (NANOSEC)	FALSE ALARM RATE (NUM/SEC)
1	10.6	220000	.72	.50	56.40	2.90	60.00	.01

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

S E N S O R S

ID	SPECTRUM	WFOV (DEGREES) HORZ	VERT	FOV SCALING	SWITCH FOV DELAY (SEC)	S/N FACTOR	POT FACTOR
1	8-12	15.00	7.80	3.00	5.00	25.702	25.073
2	8-12	13.50	6.75	3.00	5.50	0.	0.

S E N S O R   R E S O L U T I O N   C U R V E S'

MDT => INVERSE TARGET SIZE (1/MRAD) VS MDT (DEG C)
MRC => CONTRAST VS FREQUENCY (CY/MRAD)
MRT => TEMPERATURE (DEG C) VS FREQUENCY (CY/MRAD)

ID	FOV	TYPE	LIGHT LEVEL (FC)	X	Y	X	Y	X	Y	X	Y
1	NARROW	MRT	0.	0.	0.	.1000	1.6883	.2000	2.7042	.3000	3.3821
				.4000	3.8677	.5000	4.2319	.6000	4.5154	.7000	4.7422
				.8000	4.9279	1.0000	5.2138	1.5000	5.6508	2.0000	5.8980
				3.0000	6.1678	4.0000	6.3121	5.0000	6.4020	6.0000	6.4634
				7.0000	6.5080	8.0000	6.5418	9.0000	6.5684	10.0000	6.5898
2	NARROW	MRT	0.	0.	0.	.1000	1.6883	.2000	2.7042	.3000	3.3821
				.4000	3.8677	.5000	4.2319	.6000	4.5154	.7000	4.7422
				.8000	4.9279	1.0000	5.2138	1.5000	5.6508	2.0000	5.8980
				3.0000	6.1678	4.0000	6.3121	5.0000	6.4020	6.0000	6.4634
				7.0000	6.5080	8.0000	6.5418	9.0000	6.5684	10.0000	6.5898

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

T Y P E   W E A P O N S

ID	NAME	TARGET DESIGNATOR	SPEED (KM/HR)	RELOAD TIME (SECONDS)
100	M1 GUN	NONE	0	5.0
110	TOW	FIRER	350	5.0
120	HELLFIRE	SCOUT	400	5.0
130	COPPERHEAD	SCOUT	500	5.0
200	T72 GUN	NONE	0	5.0
210	HEATSEEKER	NONE	0	5.0

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

T Y P E   P L A T F O R M S

ID	NAME	JOHNSON LEVEL	OWNER ID	OWNER NAME	COLOR	PRIORITY
1	TGT	1 = DETECT	0	-	WHITE	1
2	TRACK	2 = CLASSIFY	1	TGT	WHITE	1
3	WEAPON	2 = CLASSIFY	1	TGT	RED	1
5	TANK	3 = RECOGNIZE	2	TRACK	WHITE	1
6	ATGM	3 = RECOGNIZE	3	WEAPON	RED	1
100	M1	4 = IDENTIFY	5	TANK	BLUE	1
101	ARTY UNIT	4 = IDENTIFY	1	TGT	BLUE	1
200	T72	4 = IDENTIFY	5	TANK	RED	1
201	LAUNCHER	4 = IDENTIFY	6	ATGM	RED	1
700	AIR TARGET	1 = DETECT	0	-	BLUE	1
800	ATK HELI	4 = IDENTIFY	700	AIR TARGET	BLUE	1
801	SCOUT HELI	4 = IDENTIFY	700	AIR TARGET	BLUE	1
802	RPV	4 = IDENTIFY	700	AIR TARGET	BLUE	2

---------- DATA FOR LEVEL 4 TYPE PLATFORMS ----------

ID	TYPE WEAPON ID	TYPE WEAPON NAME	CLASS	HEIGHT (METERS)	CROSS SECTION (SQ M)	DELTA T (DEG C)	REFLECTANCE	CONTRAST
100	100	M1 GUN	GROUND	2.40	7.90	1.50	.50000	0.
101	130	COPPERHEAD	GROUND	0.	0.	0.	0.	0.
200	200	T72 GUN	GROUND	2.30	6.50	1.50	.00637	0.
201	210	HEATSEEKER	GROUND	1.50	2.00	1.00	.50000	0.
800	120	HELLFIRE	AIR	1.90	3.10	2.00	.50000	0.
801	-	-	AIR	2.70	8.50	2.00	.50000	0.
802	-	-	AIR	1.00	1.50	2.00	.50000	0.

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

S I N G L E   S H O T   P R O B A B I L I T Y
O F   K I L L

--TYPE ID	WEAPON- NAME	TYPE ID	PLATFORM NAME	RANGE (METERS)	RANGED SSPK	UNRANGED SSPK
100	M1 GUN	200	T72	0	.90	.80
				500	.86	.61
				1000	.67	.41
				1500	.46	.22
				2000	.30	.10
				2500	.19	.05
				2510	0.	0.
		201	LAUNCHER	0	.90	.80
				500	.86	.61
				1000	.67	.41
				1500	.46	.22
				2000	.30	.10
				2500	.19	.05
				2510	0.	0.
110	TOW	200	T72	0	.85	.85
				4000	.75	.75
				4020	0.	0.
		201	LAUNCHER	0	.85	.85
				4000	.80	.80
				4020	0.	0.
120	HELLFIRE	200	T72	0	.85	.85
				4000	.75	.75
				4020	0.	0.
		201	LAUNCHER	0	.85	.85
				4000	.80	.80
				4020	0.	0.
130	COPPERHEAD	200	T72	0	.85	.85
				15000	.75	.75
				15020	0.	0.
		201	LAUNCHER	0	.85	.85
				15000	.80	.80
				15020	0.	0.
200	T72 GUN	100	M1	0	0.	.80
				500	0.	.61
				1000	0.	.41
				1500	0.	.22
				2000	0.	.10
				2500	0.	.05
				2510	0.	0.
210	HEATSEEKER	800	ATK HELI	0	0.	.50
				5000	0.	.40

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

S I N G L E S H O T P R O B A B I L I T Y
O F K I L L

—TYPE ID	WEAPON- NAME	TYPE ID	PLATFORM NAME	RANGE (METERS)	RANGED SSPK	UNRANGED SSPK
				5600	0.	0.
		801	SCOUT HELI	0	0.	.50
				5000	0.	.40
				5600	0.	0.
		802	RPV	0	0.	.50
				5000	0.	.40
				5600	0.	0.

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

D E F I L A D E
F R A C T I O N S   V I S I B L E

OBSERVER CLASS	TYPE ID	PLATFORM NAME	RANGE (METERS)	FRACTION VISIBLE
GROUND	100	M1	0	.30
			3000	.10
			6000	0.
GROUND	200	T72	0	1.00
			6000	1.00
GROUND	201	LAUNCHER	0	1.00
			6000	1.00
AIR	200	T72	0	1.00
			6000	1.00
AIR	201	LAUNCHER	0	1.00
			6000	1.00

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

H U N T E R - K I L L E R S

H/K ID	SIDE	TYPE ID	PLATFORM NAME	TYPE ID	WEAPON-NAME	COORDINATES X	COORDINATES Y
101	BLUE	100	M1	100	M1 GUN	500	250
102	BLUE	100	M1	100	M1 GUN	500	275
103	BLUE	100	M1	100	M1 GUN	500	300
104	BLUE	100	M1	100	M1 GUN	500	350
120	BLUE	800	ATK HELI	120	HELLFIRE	100	800
121	BLUE	800	ATK HELI	120	HELLFIRE	100	800
201	RED	200	T72	200	T72 GUN	4500	1000
202	RED	200	T72	200	T72 GUN	4600	1010
203	RED	200	T72	200	T72 GUN	4700	1020
204	RED	200	T72	200	T72 GUN	4800	1030
205	RED	200	T72	200	T72 GUN	4900	1040
206	RED	201	LAUNCHER	210	HEATSEEKER	5000	1050

	HUNTER					KILLER				
H/K ID	SENSOR	LASER	SECTOR	AZIMUTH (DEG)	JOHNSON LEVEL	SENSOR	LASER	SECTOR	AZIMUTH (DEG)	JOHNSON LEVEL
101	1	0	11	90	1=DETECT	1	1	11	90	3=RECOGNIZE
102	1	0	11	90	1=DETECT	1	1	11	90	3=RECOGNIZE
103	1	0	11	90	1=DETECT	1	1	11	90	3=RECOGNIZE
104	1	0	11	90	1=DETECT	1	1	11	90	3=RECOGNIZE
120	NONE	-	-	-	-	1	0	11	124	1=DETECT
121	NONE	-	-	-	-	1	0	11	124	1=DETECT
201	NONE	-	-	-	-	1	0	12	264	1=DETECT
202	NONE	-	-	-	-	1	0	12	264	1=DETECT
203	NONE	-	-	-	-	1	0	12	264	1=DETECT
204	NONE	-	-	-	-	1	0	12	264	1=DETECT
205	NONE	-	-	-	-	1	0	12	264	1=DETECT
206	NONE	-	-	-	-	1	0	12	264	1=DETECT

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

M O V E M E N T

INTERVAL = 100 METERS

M O V E M E N T   V E C T O R S

H/K ID	VECTOR SEQUENCE	AZIMUTH	SPEED (KM/HR)	ALTITUDE (METERS)	DISTANCE (METERS)
120	1	90	10	100	3000
201	1	264	10	-	4000
	2	270	10	-	500
202	1	264	10	-	4100
	2	270	10	-	500
203	1	264	10	-	4200
	2	270	10	-	500
204	1	264	10	-	4300
	2	270	10	-	500
205	1	264	10	-	4400
	2	270	10	-	500
206	1	264	10	-	4500
	2	270	10	-	500

Figure 4-1, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : MODEL INITIALIZATION
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

A I R   T A C T I C S

POPUP	MIN TIME	DOWN	120	SECONDS
POPUP	MAX TIME	UP	60	SECONDS
ATR	MAX TIME	UP	10	SECONDS
ATR	TGT RECOG	TIME	2	SECONDS

ACQUIRER H/K ID	ACQUIRER TYPE ID	ACQUIRER PLATFORM NAME	TACTICS	MOVE INTERVAL (METERS)	ATTACK H/K ID	ATTACK TYPE ID	ATTACK PLATFORM NAME
120	800	ATK HELI	POPUP	100	121	800	ATK HELI

Figure 4-1, (continued)

```
HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57      : MODEL INITIALIZATION
  HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

                         T E R R A I N

                      RESOLUTION IS LOW


                         C O N T R O L S

                 MU LOS               1.000 MINUTES
                 MU NLOS               .001 MINUTES


                        I N I T I A L
                  P R O B A B I L I T Y   O F
                   L I N E   O F   S I G H T

                    RANGE
                   (METERS)    PROBABILITY
                   --------    -----------
                          0       1.00
                       2000        .75
                       7000        .40
                      10000       0.
```

Figure 4-1, (continued)

4.3 REPLICATION REPORTS

The Replication Reports which appear on unit 8 are generated when controls REPLICATION.ECHO, REPORT.CONTROL, and REPORT.INTERVAL are set as described in Section 3.3.2. The five reports classified as Replication Reports are: Initial Line of Sight, Status, Scoreboard, Acquisition and Engagement Times, and Engagement Results.

4.3.1 Initial Line of Sight Report

The Initial Line of Sight Report is produced at the start of each replication when REP.ECHO is set to "YES". For each possible pair of observer platform versus target platform, the report displays which of the observer's sensors have the target in their fields of search. The possible values are: HUNTER, KILLER, H + K (hunter and killer), or NO LOS (no line of sight). This is followed by the percentage of the target that is visible to the observer. A sample of this report is shown in Figure 4-2.

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : REPLICATION 15
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

INITIAL LINE OF SIGHT FOR REPLICATION 15

DETECTING SENSORS / PERCENT VISIBLE

H/K ID	----TARGET IDS -----> 101	102	103	104	120	121
101	-	NO LOS	NO LOS	NO LOS	NO LOS	NO LOS
102	NO LOS	-	NO LOS	NO LOS	NO LOS	NO LOS
103	NO LOS	NO LOS	-	NO LOS	NO LOS	NO LOS
104	NO LOS	NO LOS	NO LOS	-	NO LOS	NO LOS
120	KILLER/ 0	KILLER/ 0	KILLER/ 0	KILLER/ 0	-	KILLER/ 0
121	KILLER/ 0	KILLER/ 0	KILLER/ 0	KILLER/ 0	KILLER/ 0	-
201	KILLER/ 6	KILLER/ 0	KILLER/ 6	KILLER/ 6	KILLER/100	KILLER/ 0
202	KILLER/ 0	KILLER/ 6	KILLER/ 6	KILLER/ 0	KILLER/100	KILLER/ 0
203	KILLER/ 0	KILLER/ 6	KILLER/ 6	KILLER/ 0	KILLER/100	KILLER/ 0
204	KILLER/ 0	KILLER/ 0	KILLER/ 5	KILLER/ 5	KILLER/100	KILLER/ 0
205	KILLER/ 5	KILLER/ 5	KILLER/ 0	KILLER/ 5	KILLER/100	KILLER/ 0
206	KILLER/ 5	KILLER/ 5	KILLER/ 5	KILLER/ 0	KILLER/100	KILLER/ 0

Figure 4-2, Initial LOS Report

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : REPLICATION 15
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

INITIAL LINE OF SIGHT FOR REPLICATION 15

DETECTING SENSORS / PERCENT VISIBLE

H/K ID	----TARGET IDS -----> 201	202	203	204	205	206
101	H + K /100	H + K /100	H + K / 0	H + K / 0	H + K /100	H + K /100
102	H + K / 0	H + K /100	H + K /100	H + K / 0	H + K /100	H + K /100
103	H + K /100	H + K /100	H + K /100	H + K /100	H + K / 0	H + K /100
104	H + K /100	H + K / 0	H + K / 0	H + K /100	H + K /100	H + K / 0
120	NO LOS	NO LOS	NO LOS	NO LOS	NO LOS	NO LOS
121	NO LOS	NO LOS	NO LOS	NO LOS	NO LOS	NO LOS
201	-	NO LOS	NO LOS	NO LOS	NO LOS	NO LOS
202	KILLER/100	-	NO LOS	NO LOS	NO LOS	NO LOS
203	KILLER/100	KILLER/100	-	NO LOS	NO LOS	NO LOS
204	KILLER/100	KILLER/100	KILLER/100	-	NO LOS	NO LOS
205	KILLER/100	KILLER/100	KILLER/100	KILLER/100	-	NO LOS
206	KILLER/100	KILLER/100	KILLER/100	KILLER/100	KILLER/100	-

Figure 4-2, (continued)

4.3.2 Status Reports

The Status Reports are produced using the two controls REPORT.CONTROL and REPORT.INTERVAL. When REPORT.CONTROL is set to 1 or 2, reports of the format shown in Figure 4-3 are generated. The reports are numbered, with the initial report in each replication being number 1. Each report is identified by this number, the number of the replication, and the simulation time at which the report was printed. For each platform the report gives:

ID	The user assigned platform ID.
SIDE	Either BLUE or RED.
COORDINATES	The x,y coordinates of the platform, in meters. For a dead platform, these are the coordinates at which it was killed.
STATUS	For a ground platform this value may be ALIVE or DEAD. For an airborne platform, this value may be MASKED, UNMASKED, or DEAD.
NUM TGTS	The number of detected targets. This may include platforms on the same side whose color cannot be distinguished. If the color can be distinguished and the target is found to be friendly, it will not be included in this statistic.
POTEN TGTS	The number of potential targets. This includes all platforms belonging to both sides that are in either the hunter or killer sensor's field of search.
AZIMUTHS	The center of each sensor's field of search, measured clockwise from 0 degrees = North. A value of zero is displayed instead of an azimuth when the platform does not have a sensor.
MOVE VECTORS	The sequence number, speed in kilometers per hour, and the direction of movement as given by the platform's current move vector. If the platform is not moving, dashes are displayed instead of numeric values.

When REPORT.CONTROL is set to 3 or 4 an additional status report is generated each time a platform is killed. This report contains the same information described above with the addition of two columns, as shown in Figure 4-4. For the platform which was the killer, the additional information displayed is:

TGT KILLED	The ID number of the platform that was killed.
RANGE	The range from the killer to the dead target, in meters.

```
HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57      : REPLICATION 15
  HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

                           STATUS   REPORT    3

                  REPLICATION 15      TIME = 15.00 MINUTES
```

H/K ID	SIDE	COORDINATES X	COORDINATES Y	STATUS	NUM TGTS	POTEN. TGTS	AZIMUTHS HNT	AZIMUTHS KLR	MOVE VECTOR INDEX	MOVE VECTOR SPEED	MOVE VECTOR AZIMUTH
101	BLUE	500	250	ALIVE	0	6	90	90	-	-	-
102	BLUE	500	275	ALIVE	0	6	90	90	-	-	-
103	BLUE	500	300	ALIVE	0	6	90	90	-	-	-
104	BLUE	500	350	ALIVE	0	6	90	90	-	-	-
121	BLUE	1000	800	DEAD	0	0	0	124	1	10	90
121	BLUE	1000	800	MASKED	0	0	0	124	-	-	-
201	RED	2025	750	ALIVE	0	5	0	264	1	10	264
202	RED	2125	760	ALIVE	0	6	0	264	1	10	264
203	RED	2225	770	ALIVE	0	7	0	264	1	10	264
204	RED	2325	780	ALIVE	0	8	0	264	1	10	264
205	RED	2425	790	ALIVE	0	9	0	264	1	10	264
206	RED	2525	800	ALIVE	0	10	0	264	1	10	264

Figure 4-3, Status Report

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : REPLICATION 15
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

STATUS REPORT 8

REPLICATION 15 TIME = 17.68 MINUTES

H/K ID	SIDE	COORDINATES X	COORDINATES Y	STATUS	NUM TGTS	POTEN. TGTS	AZIMUTHS INT	AZIMUTHS KLR	MOVE VECTOR INDEX	MOVE VECTOR SPEED	MOVE VECTOR AZIMUTH	TGT KILLED	RANGE (MET)
101	BLUE	500	250	ALIVE	0	2	90	90	-	-	-		
102	BLUE	500	275	ALIVE	0	2	90	90	-	-	-		
103	BLUE	500	300	ALIVE	0	2	90	90	-	-	-		
104	BLUE	500	350	ALIVE	0	2	90	90	-	-	-		
120	BLUE	1000	800	DEAD	0	0	0	124	1	10	90		
121	BLUE	1200	800	DEAD	0	0	0	124	-	-	-		
201	RED	1926	740	DEAD	0	0	0	264	1	10	264		
202	RED	1927	740	DEAD	0	0	0	264	1	10	264		
203	RED	1928	740	DEAD	0	0	0	264	1	10	264		
204	RED	1929	740	DEAD	0	0	0	264	1	10	264		
205	RED	2029	750	ALIVE	0	5	0	264	1	10	264		
206	RED	2129	760	ALIVE	0	6	0	264	1	10	264	121	930

Figure 4-4, Status Report After a Kill

4.3.3 The Scoreboard

The Scoreboard is produced at the end of each replication when REP.ECHO is set to "YES". This report, as shown in Figure 4-5, displays for each platform:

H/K ID	The user assigned platform ID.
SIDE	Either BLUE or RED.
TYPE OF PLATFORM	The platform's type platform ID and name.
STATUS	The plaform's status at the end of the replication (see Status Reports).
ID NUMBERS OF TARGETS KILLED	A list of the IDs of the platforms killed.
TOTAL KILLS	The number of kills made by the platform.

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : REPLICATION 15
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

SCOREBOARD FOR REPLICATION 15

H/K ID	SIDE	TYPE ID	PLATFORM NAME	STATUS	ID NUMBERS OF TARGETS KILLED	TOTAL KILLS
101	BLUE	100	M1	ALIVE	202	1
102	BLUE	100	M1	ALIVE	203	1
103	BLUE	100	M1	ALIVE	201 204 205	3
104	BLUE	100	M1	ALIVE	206	1
120	BLUE	800	ATK HELI	DEAD		0
121	BLUE	900	ATK HELI	DEAD		0
201	RED	200	T72	DEAD		0
202	RED	200	T72	DEAD		0
203	RED	200	T72	DEAD		0
204	RED	200	T72	DEAD		0
205	RED	200	T72	DEAD		0
206	RED	201	LAUNCHER	DEAD	120 121	2

Figure 4-5, The Scoreboard

4.3.4 Acquisition and Engagement Times Report

The Acquisition and Engagement Times Report is produced at the end of each replication when REPLICATION.ECHO is set to "YES". A sample report is shown in Figure 4-6. This report displays for each platform:

H/K ID — The user assigned platform ID.

ACQUISITION TIMES — The minimum, maximum, and average acquisition times, in seconds, as computed by the search routines. These statistics are only collected when the acquisition times are less than infinity. These times are given for the hunter sensor and the killer sensor, along with the minimum, maximum, and average over both sensors.

INFINITE ACQ TIMES — The number of times the search routines determined that the acquisition time would be infinite. An infinite acquisition time occurs when the number of resolvable bars is less than the user defined level for detection. Also, by definition for this model an acquisition time greater than the length of a replication is considered to be infinite.

ENGAGEMENT TIMES — The minimum, maximum, and average lengths of time the platform was firing at a target. The total time engaged is also given. The engagement duration is measured from the moment the first shot was fired to the moment when the last reload was finished. This means that an engagement is usually the weapon's reload time times the number of rounds fired.

The acquisition times must be interpreted carefully. As described in Section 2.12.4, each time a platform moves the previously scheduled acquisitions which have not yet occurred are discarded and new acquisitions scheduled. Similarly, if a target moves the acquisition time for that target must be recomputed. This is required due to the range dependence of acquisition times. Each newly computed acquisition time is included in these statistics. This explains why the number of infinite acquistions can be in the hundreds when there are only twenty platforms.

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : REPLICATION 15
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

ACQUISITION AND ENGAGEMENT TIMES FOR REPLICATION 15

H/K ID	HUNTER ACQUISITION TIME (SEC) MIN	MAX	AVG	KILLER ACQUISITION TIME (SEC) MIN	MAX	AVG	HUNTER-KILLER ACQUISITION TIME (SEC) MIN	MAX	AVG	INFINITE ACQ TIMES	ENGAGEMENT TIMES (SEC) MIN	MAX	AVG	TOTAL
101	5.00	119.31	24.98	325.27	325.27	325.27	5.00	325.27	50.00	370	5.00	10.00	6.25	25.00
102	5.00	49.88	17.19	0.	0.	0.	0.	49.88	17.19	373	10.00	10.00	10.00	10.00
103	5.00	70.14	15.99	0.	0.	0.	0.	70.14	15.99	371	5.00	5.00	5.00	25.00
104	5.00	38.77	18.75	405.53	405.53	405.53	5.00	405.53	57.42	356	5.00	5.00	5.00	10.00
120	0.	0.	0.	0.	0.	0.	0.	0.	0.	0	0.	0.	0.	0.
121	0.	0.	0.	14.02	17.67	15.85	0.	17.67	15.85	0	0.	0.	0.	0.
TOTAL BLUE	0.	119.31	12.82	0.	405.53	124.44	0.	405.53	26.08	1470	0.	10.00	4.37	70.00
201	0.	0.	0.	25.19	68.54	50.74	0.	68.54	50.74	129	0.	0.	0.	0.
202	0.	0.	0.	5.00	141.84	27.01	0.	141.84	27.01	135	0.	0.	0.	0.
203	0.	0.	0.	5.00	202.50	38.77	0.	202.50	38.77	138	0.	0.	0.	0.
204	0.	0.	0.	5.00	275.48	41.06	0.	275.48	41.06	142	0.	0.	0.	0.
205	0.	0.	0.	5.00	181.56	44.64	0.	181.56	44.64	148	0.	0.	0.	0.
206	0.	0.	0.	5.00	259.60	43.36	0.	259.60	43.36	179	5.00	5.00	5.00	10.00
TOTAL RED	0.	0.	0.	5.00	275.48	40.93	0.	275.48	40.93	871	0.	5.00	.83	10.00

Figure 4-6, Acquisition and Engagement Times Report

4.3.5 Engagement Results Report

The Engagement Results Report is produced at the end of each replication when REP.ECHO is set to "YES". A sample report is shown in Figure 4-7. This report displays for each platform:

H/K ID — The user assigned platform ID.

SHOTS — The number of shots made by the platform.

KILLS — The number of platforms killed.

LASER RANGE FINDING — If the platform has a laser range finder, the number of times range finding was attempted is displayed. In addition, the number of successful range findings is displayed.

AVG NUM POTENTIAL TARGETS — The average number of targets that were in the field of search of either of the platform's sensors.

ACQUIRED TGTS WAITING — The average and maximum number of targets that have been acquired and are waiting to be engaged. If the platform has only a killer sensor, the maximum will be one since the killer cannot acquire new targets while engaging one it has detected. The average waiting time represents the time between acquiring the target and engaging, which is when the sensor is switching from wide to narrow field of view. These times can increase if the platform has two sensors. In this case the hunter can be acquiring many targets while the killer is engaging one.

NUM ACQUIRED TGTS LOST DUE TO LOS BREAKS — Movement may cause terrain features to mask a previously detected target. Each time that occurs this statistic is increased by one.

After values for all of the platforms for a side have been displayed, the totals for the side are shown. The AVG NUM POTENTIAL TARGETS and AVG ACQUIRED TGTS WAITING values for a side are the averages over all of the side's platforms.

Two of the above statistics are weighted by time and must be interpreted carefully. These are the average number of potential targets and the average number of acquired targets waiting engagement.

SIMSCRIPT computes a time weighted average as follows:

$$\text{let SUM} = \Sigma\ (X\ (\text{TIME.V} - T_L))$$
$$\text{let AVERAGE} = \text{SUM} \ /\ (\text{TIME.V} - T_0)$$

HUNTER-KILLER MODEL VERSION 2.0 : -SEP-1986 14:57 : REPLICATION 15
HUNTER-KILLER TEST DATA - GRO' PLATFORMS PLUS 1 FLIGHT TEAM

ENGAGEMENT RESULTS FOR REPLICATION 15

H/K ID	SHOTS	KILLS	LASER RANGE FINDING TRIES	LASER RANGE FINDING SUCCESSES	AVG NUM POTENTIAL TARGETS	ACQUIRED TGTS WAITING AVG	ACQUIRED TGTS WAITING MAX	NUM ACQUIRED TGTS LOST DUE TO LOS BREAKS
101	5	1	4	1	1.78	.01	1	0
102	2	1	1	0	1.78	.02	2	0
103	5	3	5	1	1.78	.01	1	0
104	2	1	2	1	1.78	.01	1	0
120	0	0	0	0	.46	0.	0	0
121	0	0	0	0	.47	0.	0	0
TOTAL BLUE	14	6	12	3	1.34	.01	2	0
201	0	0	0	0	1.57	0.	0	0
202	0	0	0	0	1.88	0.	0	0
203	0	0	0	0	2.21	0.	0	0
204	0	0	0	0	2.53	0.	0	0
205	0	0	0	0	2.87	0.	0	0
206	2	2	0	0	3.42	.00	1	1
TOTAL RED	2	2	0	0	2.41	.00	1	1

Figure 4-7, Engagement Results Report

where TIME.V = the simulated time at which the statistic was collected

T_L = the simulated time at which the variable was set to its current value

T_0 = the simulated time at the start of the replication

X = the sample value to be included in the average

To illustrate what occurs with a time weighted statistic, assume a replication length of 30 simulated minutes. Assume that for the first 15 minutes a platform has 4 targets in its field of search. The weighted average over that time will be:

SUM = 4 (15 - 0) = 60
AVERAGE = 60 / (15 - 0) = 4

Now assume that the platform is killed. For the last 15 minutes of the replication the number of potential targets will be zero. Therefore, the value that will be included in the report which is generated at time 30 will be:

SUM = 4 (15 - 0) + 0 (30 - 15) = 60
AVERAGE = 60 / (30 - 0) = 2

Now assume that a second run is made using the same data, except that the run duration is increased to 60 minutes. The first thirty minutes of the replication will be unchanged, meaning that the platform will still be killed at 15 minutes. The final statistic included in the report, however, will change as follows:

SUM = 4 (15 - 0) + 0 (60 - 15) = 60
AVERAGE = 60 / (60 - 0) = 1

The manner in which SIMSCRIPT collects time weighted variables must be kept in mind when interpreting these two averages.

4.4 FINAL SUMMARIES

The Final Summaries which appear on unit 8 consist of six reports. These reports are always generated. The first is an Acquisition and Engagement Times Report of the same format as shown in Section 4.3.4. The only difference is that the times given are the minimums, maximums, averages, and totals over all replications. The summary Acquisition and Engagement Times Report produced by the data set from Section 3 is shown in Figure 4-8.

The second summary is an Engagement Results Report. The format is identical to that given in Section 4.3.5, with the values printed being the minimums, maximums, averages, and totals over all replications. A sample of this report is shown in Figure 4-9.

Four histograms are produced to summarize shots and losses over all replications. These histograms display a column for each replication along with two columns giving the totals and averages over all replications. The rows of the histograms are either range intervals or time intervals. These histograms are:

SHOTS BY RANGE. This report shows the number of shots made by each side at range intervals given in hectometers (hundreds of meters). For example, the interval 13 represents shots made at targets that were at a range from 1300 to 1399 meters. A sample report is shown in Figure 4-10.

SHOTS BY TIME. This report shows the number of shots made by each side at intervals of hundredths of an hour. For example, .23 represents the simulation time from .2300 hours through .2399 hours. A sample report is shown in Figure 4-11.

LOSSES BY RANGE. This report shows the number of losses on each side at range intervals given in hectometers. A sample report is shown in Figure 4-12.

LOSSES BY TIME. This report shows the number of losses on each side in time intervals of hundredths of an hour. The report also shows the relative strength of the side in each interval, computed by dividing the number of live red platforms by the number of live blue platforms. A sample report is shown in Figure 4-13.

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

ACQUISITION AND ENGAGEMENT TIMES OVER ALL REPLICATIONS

H/K	HUNTER ACQUISITION TIME (SEC)			KILLER ACQUISITION TIME (SEC)			HUNTER-KILLER ACQUISITION TIME (SEC)			INFINITE	ENGAGEMENT TIMES (SEC)			
ID	MIN	MAX	AVG	MIN	MAX	AVG	MIN	MAX	AVG	ACQ TIMES	MIN	MAX	AVG	TOTAL
101	5.00	119.31	23.99	22.26	1817.22	444.35	5.00	1817.22	52.56	5654	5.00	40.00	5.85	335.00
102	5.00	131.31	21.90	5.00	1142.51	345.79	5.00	1142.51	42.84	5705	5.00	35.00	11.33	340.00
103	5.00	177.69	24.30	5.39	929.06	306.99	5.00	929.06	43.97	5686	5.00	35.00	12.50	350.00
104	5.00	158.07	22.54	5.00	3303.99	528.53	5.00	3303.99	59.37	5716	5.00	45.00	11.92	465.00
120	0.	0.	0.	5.00	32.81	11.94	0.	32.81	11.94	0	6.57	7.50	7.41	96.30
121	0.	0.	0.	5.00	79.83	18.16	0.	79.83	18.16	0	4.01	7.50	7.04	84.50
TOTAL BLUE	0.	177.69	15.45	5.00	3303.99	275.96	0.	3303.99	38.14	22761	4.01	45.00	10.01	1670.80
201	0.	0.	0.	5.00	169.75	43.67	0.	169.75	43.67	1934	0.	0.	0.	0.
202	0.	0.	0.	5.00	250.48	42.60	0.	250.48	42.60	1993	0.	0.	0.	0.
203	0.	0.	0.	5.00	328.76	43.57	0.	328.76	43.57	2066	0.	0.	0.	0.
204	0.	0.	0.	5.00	275.48	43.46	0.	275.48	43.46	2129	0.	0.	0.	0.
205	0.	0.	0.	5.00	297.33	40.83	0.	297.33	40.83	2200	0.	0.	0.	0.
206	0.	0.	0.	5.00	344.00	42.31	0.	344.00	42.31	2436	5.00	15.00	7.78	140.00
TOTAL RED	0.	0.	0.	5.00	344.00	42.74	0.	344.00	42.74	12758	0.	15.00	1.30	140.00

Figure 4-8, Summary Acquisition and Engagement Times Report

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

ENGAGEMENT RESULTS OVER ALL REPLICATIONS

H/K ID	SHOTS	KILLS	-----LASER----- -RANGE FINDING- TRIES	SUCCESSES	AVG NUM POTENTIAL TARGETS	ACQUIRED TGTS WAITING AVG	MAX	NUM ACQUIRED TGTS LOST DUE TO LOS BREAKS
101	67	15	34	3	2.30	.01	2	0
102	68	16	30	4	2.30	.01	2	0
103	70	18	28	1	2.33	.01	2	0
104	93	26	39	2	2.36	.01	3	0
120	12	9	0	0	.69	0.	1	0
121	7	6	0	0	.71	0.	1	0
TOTAL BLUE	317	90	131	10	1.78	.01	3	0
201	0	0	0	0	1.59	0.	0	0
202	0	0	0	0	1.90	0.	0	0
203	0	0	0	0	2.23	0.	0	0
204	0	0	0	0	2.58	0.	0	0
205	0	0	0	0	2.92	0.	0	0
206	28	15	0	0	3.37	.00	1	5
TOTAL RED	28	15	0	0	2.43	.00	1	5

Figure 4-9, Summary Engagement Results Report

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

BLUE SHOTS BY RANGE

RANGE (HM)	REPLICATION NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	TOTAL	AVERAGE
3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	.07
4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	.07
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
6	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	.07
7	1	1	1	0	0	0	1	0	0	0	0	0	1	0	0	5	.33
8	1	1	0	1	1	0	1	1	1	0	0	0	0	0	0	11	.73
9	0	0	0	1	0	0	0	0	0	2	1	0	0	1	0	2	.13
10	0	2	0	3	0	1	0	0	0	0	0	1	0	0	1	9	.60
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0.
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
13	0	0	0	0	1	0	3	0	0	0	0	0	4	0	0	8	.53
14	14	13	8	11	13	7	10	8	9	25	5	9	15	8	5	160	10.67
15	9	18	3	9	5	9	11	2	9	8	10	8	6	6	6	119	7.93
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
TOTAL	25	35	12	25	21	17	26	11	19	35	17	18	27	15	14	317	21.13

Figure 4-10, Shots by Range Histogram

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

RED SHOTS BY RANGE

RANGE (NM)	REPLICATION NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	TOTAL	AVERAGE
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
4	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	3	.20
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
6	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	.07
7	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	2	.13
8	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	3	.20
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	.07
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	.07
12	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	3	.20
13	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	.07
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
15	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0	3	.20
16	0	0	1	3	0	1	0	0	1	0	2	1	0	0	1	10	.67
TOTAL	0	3	1	5	2	4	0	2	1	0	2	2	2	2	2	28	1.87

Figure 4-10, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

BLUE SHOTS BY TIME

TIME (HRS)	REPLICATION NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	TOTAL	AVERAGE
.24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
.25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
.26	6	6	3	1	2	2	6	5	6	3	2	6	2	2	1	53	3.53
.27	7	12	3	3	3	2	10	1	6	6	2	1	1	2	4	63	4.20
.28	2	7	3	3	2	4	5	1	4	5	3	3	6	4	3	55	3.67
.29	8	5	1	1	11	1	2	2	1	6	4	4	8	3	2	59	3.93
.30	1	2	1	13	2	7	2	1	1	14	4	3	9	3	1	64	4.27
.31	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	8	.53
.32	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	3	.20
.33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
.34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
.35	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	.07
.36	0	2	0	4	0	1	0	0	0	0	0	1	0	0	3	11	.73
TOTAL	25	35	12	25	21	17	26	11	19	35	17	18	27	15	14	317	21.13

Figure 4-11, Shots by Time Histogram

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

RED SHOTS BY TIME

TIME (HRS)	REPLICATION NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	TOTAL	AVERAGE
.24	0	0	1	3	0	1	0	0	1	0	2	1	0	0	1	10	.67
.25	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0	3	.20
.26	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	.07
.27	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	3	.20
.28	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	.07
.29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	.07
.30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
.31	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	3	.20
.32	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	2	.13
.33	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	.07
.34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
.35	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	3	.20
.36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
TOTAL	0	3	1	5	2	4	0	2	1	0	2	2	2	2	2	28	1.87

Figure 4-11, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

BLUE LOSSES BY RANGE

RANGE (NM)	REPLICATION NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	TOTAL	AVERAGE
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	.07
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
6	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	.07
7	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	.07
8	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	.07
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	.07
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	.07
12	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	.07
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
15	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	.07
16	0	0	1	1	0	1	0	0	1	0	1	1	0	0	1	7	.47
TOTAL	0	1	1	2	1	2	0	0	1	0	1	2	1	1	2	15	1.00

Figure 4-12, Losses By Range Histogram

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

RED LOSSES BY RANGE

RANGE (KM)	REPLICATION NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	TOTAL	AVERAGE
3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	.07
4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	.07
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
7	1	1	1	0	0	0	1	0	0	0	0	0	1	0	0	5	.33
8	1	0	0	1	1	0	1	1	1	1	0	0	0	1	0	8	.53
9	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	2	.13
10	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	3	.20
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
13	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	2	.13
14	2	2	4	4	3	1	2	3	2	5	3	2	2	3	3	41	2.73
15	2	2	1	0	0	4	2	2	3	0	2	3	2	2	2	27	1.80
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.
TOTAL	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	90	6.00

Figure 4-12, (continued)

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

BLUE LOSSES BY TIME

TIME (HRS)	REPLICATION NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	TOTAL	STRENGTH AVERAGE	(RED/BL)
.24	0	0	1	1	0	1	0	0	1	0	1	1	0	0	1	7	.47	1.084
.25	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	.07	1.098
.26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.	.951
.27	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	.07	.741
.28	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	.07	.575
.29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	.07	.405
.30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.	.190
.31	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	.07	.115
.32	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	.07	.078
.33	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	.07	.079
.34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.	.079
.35	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	.07	.067
.36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.	0.
TOTAL	0	1	1	2	1	2	0	0	1	0	1	2	1	1	2	15	1.00	

Figure 4-13, Losses by Time Histogram

HUNTER-KILLER MODEL VERSION 2.0 : 16-SEP-1986 14:57 : FINAL REPORTS
HUNTER-KILLER TEST DATA - GROUND PLATFORMS PLUS 1 FLIGHT TEAM

RED LOSSES BY TIME

TIME (HRS)	REPLICATION NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	TOTAL	STRENGTH AVERAGE	(RED/BL)
.24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.	1.084
.25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.	1.098
.26	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	12	.80	.951
.27	2	2	1	1	1	1	2	1	1	1	1	1	1	1	1	18	1.20	.741
.28	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	14	.93	.575
.29	1	1	1	1	0	1	1	1	1	1	1	2	0	1	1	14	.93	.405
.30	1	1	1	1	2	1	1	1	1	1	1	1	2	1	1	17	1.13	.190
.31	1	0	0	0	0	0	1	1	1	1	0	0	0	1	0	6	.40	.115
.32	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	3	.20	.078
.33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.	.079
.34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.	.079
.35	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	.07	.067
.36	0	1	0	1	0	1	0	0	0	0	0	1	0	0	1	5	.33	0.
TOTAL	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	90	6.00	

Figure 4-13, (continued)

4.5 DEBUGGING REPORTS

The Debugging Reports which appear in unit 6 may be divided into five types, three of which are under the control of the user. The messages printed by the model are all of the form:

nn tt.tttt rrrrrr message

where nn	is the number of the replication
tt.tttt	is the elapsed simulation time from the beginning of the replication, in decimal hours
rrrrrr	is the first six characters of the name of the routine which produced the message
message	is the message.

The first type of Debugging Report is generated by every run and has a format as shown in Figure 4-14. It is intended to allow the user to quickly determine that a run executed as planned.

The generation of the next three types of Debugging Reports are under the control of the user. They are intended to allow the maintenance programmer to examine the actions of the model in great detail. Turning on any of these reports will cause a huge amount of data to be written to unit 6. Since these messages are intended for debugging purposes, the exact information printed is subject to frequent change to suit the purposes of the maintenance programmer. Due to the changability of these reports, samples of the messages printed are shown but are not discussed in detail.

Setting GENERAL.TRACE to "YES" will cause a message to be printed each time the model performs a significant action. A sample of these messages is shown in Figure 4-15.

Setting SEARCH.TRACE to "YES" will cause a message to be printed each time the search routines are called. A sample of these messages is shown in Figure 4-16.

Setting TERRAIN.TRACE to "YES" will cause a message to be printed each time the line of sight computations are performed. A sample of these messages is shown in Figure 4-17.

Finally, if an error causes a run to abort, a traceback will be printed. The format of this report is under the control of SIMSCRIPT II.5. If this occurs, refer the data set that was used and all generated reports to the maintenance programmer for further action.

```
-- --.---- MAIN   RUN STARTED ON 16-SEP-1986 AT 14:57
-- --.---- MAIN   DATA HAVE BEEN READ
-- --.---- MAIN   DATA HAVE BEEN ECHOED
 1 0.     MAIN   START REPLICATION  1
 1 1.0000 MAIN   END   REPLICATION  1
 2 0.     MAIN   START REPLICATION  2
 2 1.0000 MAIN   END   REPLICATION  2
 3 0.     MAIN   START REPLICATION  3
 3 1.0000 MAIN   END   REPLICATION  3
 4 0.     MAIN   START REPLICATION  4
 4 1.0000 MAIN   END   REPLICATION  4
 5 0.     MAIN   START REPLICATION  5
 5 1.0000 MAIN   END   REPLICATION  5
 6 0.     MAIN   START REPLICATION  6
 6 1.0000 MAIN   END   REPLICATION  6
 7 0.     MAIN   START REPLICATION  7
 7 1.0000 MAIN   END   REPLICATION  7
 8 0.     MAIN   START REPLICATION  8
 8 1.0000 MAIN   END   REPLICATION  8
 9 0.     MAIN   START REPLICATION  9
 9 1.0000 MAIN   END   REPLICATION  9
10 0.     MAIN   START REPLICATION 10
10 1.0000 MAIN   END   REPLICATION 10
11 0.     MAIN   START REPLICATION 11
11 1.0000 MAIN   END   REPLICATION 11
12 0.     MAIN   START REPLICATION 12
12 1.0000 MAIN   END   REPLICATION 12
13 0.     MAIN   START REPLICATION 13
13 1.0000 MAIN   END   REPLICATION 13
14 0.     MAIN   START REPLICATION 14
14 1.0000 MAIN   END   REPLICATION 14
15 0.     MAIN   START REPLICATION 15
15 1.0000 MAIN   END   REPLICATION 15
-- --.---- MAIN   HUNTER-KILLER MODEL SUCCESSFULLY COMPLETED
```

Figure 4-14, Standard Messages

```
 1   .3014 ENGAGE HK=104         - DELAY TO CHANGE TO NARROW FOV
 1   .3023 TARGET HK=206 TGT=205 - KILLER DROPS FRIENDLY TARGET
 1   .3028 ASSESS HK=104 TGT=205 - SHOOT & KILL, RANGE= 1482 PK= .23
 1   .3035 POPUP. HK=120 - POPUP IN PLACE AFTER AN ENGAGEMENT
 1   .3035 UNMASK HK=120 - UNMASK PLATFORM
 1   .3056 RESUME HK=104 TGT=206 - KILLER ACQUIRES IN7.2370E+75 SEC
 1   .3056 ENGAGE HK=104         - KILLER RESUMES SEARCHING
 1   .3123 AIR.SE SC=120 ATT=120 DES=SCOUT - START AIR ENGAGEMENT
 1   .3123 AIR.EN SC=120 ATT=120 TGT=206 RANGE TO ATTACK PLAT=  833
 1   .3144 AIR.EN SC=120 TGT=206 - STATUS = 4 WHEN ROUND AT TARGET
 1   .3144 REMASK HK=120 - REMASKS
 1   .3144 BREAK. HK=206 TGT=120 - ACQUISITION CANCELED
 1   .3144 ASSESS HK=120 TGT=206 - SHOOT & KILL, RANGE=  833 PK= .84
```

Figure 4-15, General Trace Output

```
1    .2642 SEARCH HK=103 TGT=204 SNSR=  1 SCTR= 11 RANGE= 1789 PCT=100
                  FOV=1  AREA=2  DSRD= 3  ACHV= 0  TIME=7.237E+75 SEC
1    .2642 SEARCH HK=103 TGT=205 SNSR=  1 SCTR= 11 RANGE= 1888 PCT=100
                  FOV=1  AREA=2  DSRD= 3  ACHV= 0  TIME=7.237E+75 SEC
1    .2642 SEARCH HK=103 TGT=206 SNSR=  1 SCTR= 11 RANGE= 1987 PCT=100
                  FOV=1  AREA=2  DSRD= 3  ACHV= 0  TIME=7.237E+75 SEC
1    .2667 SEARCH HK=201 TGT=120 SNSR=  1 SCTR= 12 RANGE=  828 PCT=100
                  FOV=1  AREA=2  DSRD= 1  ACHV= 2  TIME=  52.0327 SEC
1    .2667 SEARCH HK=120 TGT=201 SNSR=  1 SCTR= 11 RANGE=  828 PCT=100
                  FOV=1  AREA=2  DSRD= 1  ACHV= 2  TIME=   5.0000 SEC
1    .2667 SEARCH HK=202 TGT=120 SNSR=  1 SCTR= 12 RANGE=  927 PCT=100
                  FOV=1  AREA=2  DSRD= 1  ACHV= 2  TIME=  39.6872 SEC
1    .2667 SEARCH HK=203 TGT=120 SNSR=  1 SCTR= 12 RANGE= 1027 PCT=100
                  FOV=1  AREA=2  DSRD= 1  ACHV= 2  TIME=  13.3473 SEC
1    .2667 SEARCH HK=204 TGT=120 SNSR=  1 SCTR= 12 RANGE= 1126 PCT=100
                  FOV=1  AREA=2  DSRD= 1  ACHV= 2  TIME= 121.1107 SEC
1    .2667 SEARCH HK=205 TGT=120 SNSR=  1 SCTR= 12 RANGE= 1226 PCT=100
                  FOV=1  AREA=2  DSRD= 1  ACHV= 1  TIME=7.237E+75 SEC
1    .2667 SEARCH HK=206 TGT=120 SNSR=  1 SCTR= 12 RANGE= 1326 PCT=100
                  FOV=1  AREA=2  DSRD= 1  ACHV= 1  TIME=  18.9028 SEC
```

Figure 4-16, Search Trace Output

```
1    .2900 LINE.0 TERR RESOLUTION=LOW  OBS=206 TGT=120 PCT VIS=  0
1    .2900 LINE.0 TERR RESOLUTION=LOW  OBS=206 TGT=121 PCT VIS=  0
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=101 TGT=205 PCT VIS=100
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=205 TGT=101 PCT VIS= 20
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=102 TGT=205 PCT VIS=100
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=205 TGT=102 PCT VIS= 20
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=103 TGT=205 PCT VIS=100
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=205 TGT=103 PCT VIS= 20
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=104 TGT=205 PCT VIS=100
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=205 TGT=104 PCT VIS= 20
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=120 TGT=205 PCT VIS=  0
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=205 TGT=120 PCT VIS=  0
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=121 TGT=205 PCT VIS=  0
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=205 TGT=121 PCT VIS=  0
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=206 TGT=205 PCT VIS=100
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=101 TGT=206 PCT VIS=100
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=206 TGT=101 PCT VIS= 19
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=102 TGT=206 PCT VIS=100
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=206 TGT=102 PCT VIS= 19
1    .3000 LINE.0 TERR RESOLUTION=LOW  OBS=103 TGT=206 PCT VIS=100
```

Figure 4-17, Terrain Trace Output

This page was intentionally left blank.

5.0 OPERATING PROCEDURES

5.1 OVERVIEW

The Hunter-Killer Model is a small digital computer simulation and requires minimal resources for operation. The object (executable) program and the run stream reside on on-line storage devices. The program is normally executed through a batch job submission to the operating system from a remote terminal, but may be executed interactively.

The Hunter-Killer Model has been installed on two computers: MERADCOM's CDC Cyber 835 and NV&EOC's IBM 4341. The CDC uses the NOS/BE operating system and the IBM uses the CMS operating system. Run submission on each of these systems will be discussed in this section. The user is assumed to have a basic knowledge of the operating systems.

The model requires up to four files, two for input and two for output. These files are as follows:

DATA — An input file containing all data except the high resolution terrain elevations. This file will be read from unit 3.

TERRAIN — An input file containing the elevations used for high resolution terrain. If low resolution terrain is used this file will not appear. This file will be read from unit 4.

MESSAGES — An output file containing messages printed at the start and end of each replication. If the trace switches are set to YES, the trace messages will also appear in this file. If an error causes the run to abort, the traceback will appear here. This file will be written to unit 6.

REPORTS — An output file containing all generated reports except those in MESSAGE. This file will be written to unit 8.

5.2 USING THE CYBER 835

This section describes how to submit a job on the Cyber 835 using the NOS/BE operating system. Further information on the commands used can be found in the *INTERCOM Version 5 Reference Manual* and the *NOS/BE Version 1 Reference Manual*. The *MERADCOM Management Information Systems Directorate Computer Center User's Guide* gives more information on their standards for setting up job and task cards.

5.2.1 Creating a CDC Run Stream

The Hunter-Killer Model can require five to thirty minutes of clock time to execute. The time is dependent on the data used, such as the number of replications and the number of platforms, which affect the amount of computer time required. It is also dependent on how busy the computer is at the time the job is submitted. The amount of clock time required can be several times the amount of computer time used. If a job is submitted interactively, the terminal cannot be used for any other task until the interactive job has been completed. To avoid tying up the terminal unnecessarily, the Hunter-Killer Model should be submitted as a batch job.

To execute the Hunter-Killer Model, use the editor to create the following file. It is assumed that the executable module has the name HK20ABS. The creation of the executable module is described in the Programmer's Manual.

```
xxaab,T600.
TASK(TNxxnnnnn,PW#####,TRTS) name
ATTACH,A,DATA,ID=xxnnnnn.
ATTACH,B,TERRAIN,ID=xxnnnnn.
ATTACH,ABS,HK20ABS,ID=xxnnnnn.
ABS,SIMU3=A,SIMU4=B,SIMU6=C,SIMU8=D.
CATALOG,C,MESSAGE,ID=xxnnnnn,RP=998.
CATALOG,D,REPORTS,ID=xxnnnnn,RP=998.
EXIT,S.
CATALOG,C,MESSAGE,ID=xxnnnnn,RP=998.
CATALOG,D,REPORTS,ID=xxnnnnn,RP=998.
```

Figure 5-1, CDC Run Stream with High Resolution Terrain

The first record is the job card. The field "xx" should be replaced by the first two characters of the user's task ID. The "aa" should be replaced by the user's initials. The "b" should be replaced by a unique identifier for the job. (These are standards required by MERADCOM.) The "T600" allows the job 600 octal seconds (384 decimal) in which to run.

The second record is the task card. The "xxnnnnn" should be replaced by the user's task ID. The "#####" should be replaced by the user's password. The "name" field should be replaced by the user's name.

The keywords DATA and TERRAIN should be replaced with the names of the files containing the data. The keywords MESSAGE and REPORTS should be changed to the names under which these files are to be catalogued.

All occurrences of "xxnnnnn" on the remaining records should be replaced by the IDs where the data files are stored and where the output files should be cataloged. These do not all need to be the same. In particular, there may be only one copy of the executable module, HK20ABS, stored on the maintenance programmer's account. The ID on the record that ATTACHes that module should then be the maintenance programmer's ID. Similarly, there may be only one copy of the terrain data since that will seldom change.

The first set of CATALOG commands are executed if the job completes normally. If the job aborts, the run stream skips to an EXIT,S command and performs the commands that follow it. In order to save the output in case of an error, the second set of CATALOG commands is required after the EXIT,S.

If low resolution terrain is used, the run stream should be modified to remove reference to high resolution terrain as follows:

```
xxaab,T600.
TASK(TNxxnnnnn,PW#####,TRTS) name
ATTACH,A,DATA,ID=xxnnnnn.
ATTACH,ABS,HK20LGO,ID=xxnnnnn.
ABS,SIMU3=A,SIMU6=C,SIMU8=D.
CATALOG,C,MESSAGE,ID=xxnnnnn,RP=998.
CATALOG,D,REPORTS,ID=xxnnnnn,RP=998.
EXIT,S.
CATALOG,C,MESSAGE,ID=xxnnnnn,RP=998.
CATALOG,D,REPORTS,ID=xxnnnnn,RP=998.
```

Figure 5-2, Sample CDC Run Stream, Low Resolution Terrain

After creating the run stream it should be saved in a local file, such as RUNPROC. The run stream may then be executed. Since the file contains the password for the account, it should not be permanently catalogued.

5.2.2 Submitting a Batch Run Stream

Once the run stream shown in section 5.2.1 has been created and saved in file RUNPROC, the procedure must be submitted. This is accomplished through the following commands:

```
REWIND,RUNPROC
COPY,RUNPROC,A
REWIND,A
BATCH,A,INPUT,HERE
```

Submitting a batch run stream causes that file to be deleted from the user's terminal files. If file RUNPROC is submitted and the job must be run again - such as when an error in the data causes the job to abort - file RUNPROC would need to be recreated. Copying the run stream to a temporary file and submitting that temporary file, as shown above, allows RUNPROC to remain available.

The HERE option of the BATCH command causes the dayfile to be held for the terminal instead of printed when the job finishes. MINE may be used in place of HERE (see the CDC INTERCOM 5 reference for exact usage). If neither option is used, the dayfile will automatically be sent to the printer.

The operating system will assign a name to the job of the form "xxaabyy", where "xxaab" is taken from the job card and "yy" is assigned by the system. A message will be displayed as follows:

xxaabyy SENT TO INPUT QUEUE

The progress of that job may be followed in several ways using that jobname. The following commands will show the job's status:

- Q,I This command will display a list of all jobs in the input queue waiting to be executed.
- Q,E This command displays a list of all jobs that are executing.
- Q,O This command displays a list of all jobs that have finished executing and are in the output queue.
- MYQ This command displays a list of the user's jobs in all of the queues. Only the user's jobs will be displayed.
- FILES This command lists all files assigned to the terminal. If the job is executing it will be shown under the heading "--REMOTE EXECUTING JOBS--". If the job has been completed, it will be shown under the heading "--REMOTE OUTPUT FILES--".

5.2.3 Retrieving a Batch Job

Once the batch job has been completed, the output can be examined. The output consists of the report files CATALOGed by the run stream plus a dayfile. The dayfile is a log that shows all commands that were executed and the amount of time and memory used by the job. This dayfile has the same name as the job, "xxaabyy". This dayfile is the file shown when the command Q,O is used or when the FILES command shows a remote output file. To examine this file, use the command:

BATCH,xxaabyy,LOCAL

The file will now be a local file that can be examined using the editor. The dayfile should always be BATCHed to the terminal and printed or deleted. The Computer Center will print all dayfiles that are left on the output queue more than a few days.

All of the output files produced by the model were saved by the run procedure. They may be ATTACHed to the terminal and examined using the editor or may be sent to the printer. Samples of all reports are shown in Section 4.

5.3 USING THE IBM 4341

5.3.1 Sample IBM Run Procedure

A run procedure has been developed for the Hunter-Killer Model for use on an IBM computer running under CMS. This procedure is shown in Figure 5-3. This procedure requires the file names to be of specific types, as described below. These requirements may be modified to suit the needs of a particular user.

This procedure assumes all files pertaining to the Hunter-Killer Model are stored on the same disk, referred to by the token &DISK. The procedure in the figure assumes the files will be on the user's A disk. To use another disk, modify the command:

```
&DISK = A
```

The procedure assumes that the executable module is in file HKMODEL MODULE &DISK. The creation of this module is described in the Programmer's Manual.

This procedure requires that the data set used by this model have a file type of DATA; for example, DDDDDDDD DATA &DISK, where DDDDDDDD is any name of up to eight characters. The high resolution terrain data must have a file type of TERRAIN; for example TTTTTTTT TERRAIN &DISK, where TTTTTTTT is any name of up to eight characters.

In order to relate the report files to the data set that produced them, the report files all have the same name as the data set that was used. Different file types will be used to distinguish the files. The files created are:

> **DDDDDDDD MESSAGES &DISK** will contain the MESSAGE output described in Section 5.1.
>
> **DDDDDDDD REPORTS &DISK** will contain the REPORTS output described in Section 5.1.
>
> **DDDDDDDD ERRORS &DISK** will be created only if a SIMSCRIPT error was detected. This file will contain the traceback. On the CDC computer, the traceback would have been included in the MESSAGES file; however, IBM always uses a separate file for the traceback.

Assume the procedure is in a file named HKRUN EXEC &DISK. Then, to execute the procedure, enter the command:

```
HKRUN DDDDDDDD TTTTTTTT
```

If low resolution terrain is used, omit the name of the terrain data set:

```
HKRUN DDDDDDDD
```

The procedure checks for the existance of the data sets. If they do not exist, an error message is displayed and the procedure stops.

5.3.2 Warnings Concerning the IBM

5.3.2.1 Quote Marks

Upon logging on to the IBM system, four logical line editing characters are defined. The default logical escape character is " (the quote mark). SIMSCRIPT uses the quote mark to delimit text strings in the source code. Additionally, the data sets use pairs of quote marks to delimit the names of type platforms and type weapons. To enter quote marks into the source code and in the data set using the CMS editor it is necessary to redefine the logical escape character. To determine the current definition, issue the CP command:

```
QUERY TERMINAL
```

To change the logical escape character to a new character such as the caret, enter the following CP command:

```
TERMINAL ESCAPE
```

Alternately, turn off the logical escape character using the CP command:

```
TERMINAL ESCAPE OFF
```

5.3.2.2 Memory

Upon logging on to the IBM system, a user should have 760K of storage available. This may be verified with the CP command:

```
QUERY STOR
```

Depending on the number of platforms in the data set, the number of replications being made, and whether high or low resolution terrain is used, the Hunter-Killer Model may not run within this default amount of memory. If a run aborts due to insufficient memory, the system displays one of two messages, depending on whether the operating system or SIMSCRIPT II.5 determined that all available memory had been used. The message generated by the operating system is displayed at the terminal and has the form:

```
DMSABN155T USER ABEND 0016
```

The message generated by SIMSCRIPT II.5 is written to the ERRORS file as part of a traceback. The message is:

```
***** SIMSCRIPT II.5 ERROR 2016

****NO MEMORY SPACE AVAILABLE
```

The user must then increase the memory assigned to the terminal. It will then be necessary to reinitialize CMS. The CP commands to do this are:

```
DEF STOR 1024K
IPL CMS
```

5.3.2.3 Loader Tables

Upon logging on to the IBM system, a user should have three loader tables available. This may be verified with the CP command:

QUERY LDRTBLS

A loader table is used by the system when referencing certain variables in a model. Due to the number of variables used, the Hunter-Killer Model requires that the user have 4 loader tables. If there are an insufficient number of tables, a run will abort and the system will display the following message:

DMSMOD116S LOADER TABLE OVERFLOW

In order to insure a sufficient number of tables, the following CMS command is included in the run procedure in Figure 5-3:

SET LDRTBLS 4

If the run procedure in Figure 5-3 is not used, this command will still need to be issued to allow the model to run.

```
*PROCEDURE HKRUN EXEC
*
*THIS PROCEDURE EXECUTES THE HUNTER-KILLER MODEL.
*
*ALL FILES USED BY THIS RUN MUST BE ON THE SAME DISK, IDENTIFIED IN THIS
*PROCEDURE BY THE TOKEN &DISK.  ALL OUTPUT WILL BE WRITTEN TO THAT SAME
*DISK.
*
*THE INPUT FILES USED BY THIS PROCEDURE ARE:
*                              INPUT
*          FILE                UNIT          CONTENTS
*
*   ------------------------  ------  --------------------------------
*   (DATAFILE) DATA    &DISK   SIMU03  ALL DATA EXCEPT THE HIGH
*                                      RESOLUTION TERRAIN ELEVATIONS.
*   (FILE2)    TERRAIN &DISK   SIMU04  THE HIGH RESOLUTION TERRAIN
*                                      DATA.  THIS FILE IS ONLY USED
*                                      WHEN CONTROL TERR.RESOLUTION IS
*                                      SET TO "HIGH".
*
*THE OUTPUT FILES HAVE THE SAME NAME AS (DATAFILE).  THESE FILES ARE:
*                               OUTPUT
*          FILE                 UNIT       CONTENTS
*
*   -------------------------  ------  --------------------------------
*   (DATAFILE) MESSAGES &DISK   SIMU06  SYSTEM MESSAGES AND TRACE
*   (DATAFILE) REPORTS  &DISK   SIMU08  THE REPORTS
*   (DATAFILE) ERRORS   &DISK   -       TRACEBACK FROM SIMSCRIPT
*                                       EXECUTION ERRORS
*
*TO EXECUTE THIS PROCEDURE ENTER:
*   FOR LOW  RESOLUTION TERRAIN => HKRUN (DATAFILE)
*   FOR HIGH RESOLUTION TERRAIN => HKRUN (DATAFILE) (FILE2)
*
*FIRST, TURN OFF ALL MESSAGES
&CONTROL OFF NOMSG
*
*NEXT, INCREASE THE NUMBER OF LOADER TABLES FROM THE DEFAULT VALUE OF 3
SET LDRTBLS 4
*
*NEXT, SET THE DEFAULT DISK FOR ALL HUNTER-KILLER MODEL FILES
&DISK = A
```

Figure 5-3, IBM CMS Run Stream

```
*
*NEXT, CHECK FOR CORRECT INVOCATION FORMAT
&IF &INDEX GT 0 &GOTO -DATAFILE
&BEGTYPE

 ERROR: NO DATA FILE NAME WAS GIVEN

 THE FORMAT OF THE COMMAND IS: HKRUN DATAFILE TERRAIN
    WHERE DATAFILE IS THE NAME OF THE FILE CONTAINING THE DATA.  THE
                   FILE TYPE MUST BE DATA.
          TERRAIN  IS THE NAME OF THE FILE CONTAINING THE HIGH
                   RESOLUTION TERRAIN DATA.  THE FILE TYPE MUST BE
                   TERRAIN.  THIS FIELD SHOULD BE BLANK WHEN LOW
                   RESOLUTION TERRAIN IS USED.

&END
&EXIT
*
*NEXT, DEFINE THE FILE CONTAINING THE DATA
-DATAFILE &CONTINUE
STATE &1 DATA &DISK
&IF &RETCODE = 0 &GOTO -FIDATA
&TYPE
&TYPE ERROR: INVALID DATA FILE NAME OF "   &1   " WAS GIVEN
&TYPE
&EXIT
-FIDATA &CONTINUE
FI SIMU03 DISK &1 DATA &DISK (RECFM FB LRECL 80
*
*NEXT, DEFINE THE HIGH RESOLUTION TERRAIN FILE
&IF &INDEX LT 2 &GOTO -ERASE
STATE &2 TERRAIN &DISK
&IF &RETCODE = 0 &GOTO -FITERR
&TYPE
&TYPE ERROR: INVALID HIGH RES. TERRAIN DATA FILE NAME OF "   &2   "
&TYPE
&EXIT
-FITERR &CONTINUE
FI SIMU04 DISK &2 TERRAIN &DISK (RECFM FB LRECL 80
*
*NEXT, DELETE OLD COPIES OF THE OUTPUT FILES
-ERASE &CONTINUE
ERASE &1 MESSAGES &DISK
ERASE &1 REPORTS  &DISK
ERASE &1 ERRORS   &DISK
*
*NEXT, DEFINE THE OUTPUT FILES
FI SIMU06   DISK &1 MESSAGES &DISK (RECFM FBA LRECL  80
FI SIMU08   DISK &1 REPORTS  &DISK (RECFM FBA LRECL 133
FI ERRTRACE DISK &1 ERRORS   &DISK (RECFM FB  LRECL 133
```

Figure 5-3, Continued

```
*
*NEXT, DISPLAY A PROGRESS MESSAGE
&BEGTYPE
************************************************************************
BEGIN EXECUTION OF THE HUNTER-KILLER MODEL *****************************
&END
*
*NEXT, EXECUTE THE MODEL
HKMODEL
*
*NEXT, PRINT A PROGRESS MESSAGE
&IF &RETCODE = 0 &GOTO -ENDMSG
&BEGTYPE
SIMSCRIPT EXECUTION ERROR DETECTED
&END
-ENDMSG &CONTINUE
&BEGTYPE
END   EXECUTION OF THE HUNTER-KILLER MODEL *****************************
************************************************************************
&END
*
*LAST, CLEAR ALL FILE DEFINITIONS
FI * CLEAR
*
*END PROCEDURE HKRUN EXEC
```

Figure 5-3, Continued

5.4 RUN ERRORS

If an error occurs during the execution of the model, the MESSAGE file will not end with the line:

-- --.---- MAIN HUNTER-KILLER MODEL SUCCESSFULLY COMPLETED

SIMSCRIPT will generate a traceback. On CDC, the traceback will appear in the MESSAGE file. On IBM, the traceback will appear in the separate ERRORS file described in Section 5.3. In either case, the file will show the routine in which the error occurred. If the error was in a read routine, examine the data file in the data block whose name corresponds to the read routine name to determine if there was a format error. If there was no format error or if the error did not occur in a read routine, refer the data set used and all output that was generated to the maintenance programmer.

This page was intentionally left blank.

APPENDIX A - REFERENCES

C.A.C.I., Inc. 1983. *SIMSCRIPT II.5 Programming Language*. Los Angeles: C.A.C.I.

C.A.C.I., Inc. 1983. *SIMSCRIPT II.5 Reference Handbook, Second Edition*. Los Angeles: C.A.C.I.

C.A.C.I., Inc. 1984. *SIMSCRIPT II.5 User's Manual, IBM S/370 Series*. Los Angeles: C.A.C.I.

C.A.C.I., Inc. 1985. *SIMSCRIPT II.5 User's Manual, CDC Release 5*. Los Angeles: C.A.C.I.

Control Data Corporation. 1982. *CYBER Loader Version 1 Reference Manual*. CDC publication 60429800. Sunnyvale: Control Data Corporation.

Control Data Corporation. 1982. *INTERCOM Version 5 Reference Manual*. CDC publication 60455010. St. Paul: Control Data Corporation.

Control Data Corporation. 1982. *NOS/BE Version 1 Reference Manual*. CDC publication 60493800. St. Paul: Control Data Corporation.

Defense Mapping Agency. 1983. *Product Specifications for Digital Landmass System (DLMS) Data Base, Second Edition*. DMA publication PS/1CD/200, PS/1CE/200, PS/1CF/200, PS/1CG/200. Stock number SPEXDLMS2.

Fairbrother, Ross and Campbell, Robert T. 1986. *Hunter-Killer Model Database System Programmer's Manual*. Unpublished document prepared for the Night Vision and Electro-Optics Center by Science Applications International Corporation.

Fairbrother, Ross and Campbell, Robert T. 1986. *Hunter-Killer Model Database System User's Manual*. Unpublished document prepared for the Night Vision and Electro-Optics Center by Science Applications International Corporation.

Fairbrother, Ross and Campbell, Robert T. 1986. *Hunter-Killer Model Version 2.0 Executive Summary*. Unpublished document prepared for the Night Vision and Electro-Optics Center by Science Applications International Corporation.

Fairbrother, Ross and Campbell, Robert T. 1986. *Hunter-Killer Model Version 2.0 Programmer's Manual*. Unpublished document prepared for the Night Vision and Electro-Optics Center by Science Applications International Corporation.

Fairbrother, Ross and Campbell, Robert T. 1986. *Hunter-Killer Model Version 2.0 User's Manual*. Unpublished document prepared for the Night Vision and Electro-Optics Center by Science Applications International Corporation.

IBM. 1983. *Virtual Machine/System Product CMS Command and Macro Reference, Release 3*. IBM publication SC19-6209. Endicott: IBM.

IBM. 1983. *Virtual Machine/System Product CMS User's Guide, Release 3.* IBM publication SC19-6210-2. Endicott: IBM.

IBM. 1983. *Virtual Machine/System Product Editor Command and Macro Reference, Release 3.* IBM publication SC24-5221-2. Endicott: IBM.

IBM. 1983. *Virtual Machine/System Product Editor User's Guide, Release 3.* IBM publication SC24-5220-1. Endicott: IBM.

IBM. 1983. *VS FORTRAN Application Programming Language Reference, Release 3.0.* IBM publication GC26-3986-3.

Lawson, W.R.; Cassidy, T.W.; and Ratches, J.A. 1978. A Search Model. (UNCLASSIFIED) Presented at the IRIS Specialty Group on Imaging, June 1978.

MERADCOM. 1982. *Management Information Systems Directorate Computer Center User's Guide.* Fort Belvoir: MERADCOM.

NV&EOL. 1985. *NVEOL Target Acquisition Performance Handbook - Vol. 2 (U). Final Draft.* (CONFIDENTIAL) U.S. Army Electronics R&D Command, Night Vision and Electro-Optics Laboratory. September, 1985.

Obert, L.P.; Wood, J.; and Nash, C. 1981. *Visionics E-O Sensor Performance Handbook - Vol. 1 - Natural European Environments (U).* (CONFIDENTIAL) U.S. Army Electronics R&D Command, Night Vision and Electro-Optics Laboratory. June, 1981.

Ratches, J.A.; Lawson, W.R.; Obert, L.P.; Bergemann, R.J.; Cassidy, T.W.; and Swenson, J.M. 1975. Night Vision Laboratory Static Performance Model for Thermal Viewing Systems. (UNCLASSIFIED) U.S. Army Electronics R&D Command, Night Vision and Electro-Optics Laboratory Report ECOM-7034.

Ratches, J.A.; Lawson, W.R.; Shields, F.J.; Hoover, C.W.; Obert, L.P.; Rodak, S.P.; and Sola, M.C. 1981. *Status of Sensor Performance Modeling at NV&EOL.* (UNCLASSIFIED) Unpublished paper of the U.S. Army Electronics R&D Command, Night Vision and Electro-Optics Laboratory. May 1, 1981.

RCA Corporation, Solid State Division, Electro Optics and Devices. 1974. *Electro-Optics Handbook.* Technical Series EOH-11. Lancaster: RCA.

Russell, Edward C. 1983. *Building Simulation Models with SIMSCRIPT II.5.* Los Angeles: C.A.C.I.

Sola, M.C.; Hoover, C.W.; Stamboulas, N.; and Fleury, J. 1983. *E-O Sensors for the FADEW Study (U).* (SECRET/NOFORN) U.S. Army Electronics R&D Command, Night Vision and Electro-Optics Laboratory Official Draft. April, 1983.

Vollmerhauser, Richard; Nash, Carolyn; Kessler, Howard. 1986. *Advanced Processor/FLIR Performance Measures (U).* (SECRET) Unpublished paper of the U.S. Army Electronics R&D Command, Night Vision and Electro-Optics Center. March 11, 1986.

APPENDIX B - TERMS AND ABBREVIATIONS

(Note: The source of many of the following definitions is the *Visionics E-O Sensor Performance Handbook.*)

Aerosol: Solid or liquid particles dispersed in a gas. Atmospheric hazes and fogs, smokes, and other particulate pollutants can be regarded as aersols. To calculate the extinction of radiation by an aerosol, the aerosol must be described by its particle size distribution, particle number density, and the complex indexes of refraction of its components.

Aerosol Extinction: The atmospheric attenuation of radiation due to scattering and absorption by suspended aerosol particles.

Apparent Contrast: The target-to-background contrast seen by an observer or other sensor separated from the target scene by a contrast degrading medium, such as the atmosphere.

Atmospheric Transmittance: The fraction of the target radiance which reaches the target acquisition sensor. Atmospheric transmittance is less than 1.0 due to molecular and aerosol extinction.

Attenuation (or Extinction): The removal of energy by scattering and absorption from radiation traversing a medium.

Automatic Target Recognition (ATR) Sensor: A sensor capable of interpreting a signal to determine if a target is present.

Battlefield Identification Friend or Foe (BIFF): The ability to interrogate an electronic emission from a source and through signal analysis determine whether the source is a friend (on the same side) or a foe. This is independent of a sensor's ability to acquire the source at any given level.

Classification: The ability to distinguish a target by general types; e.g., as a tracked vehicle instead of a wheeled vehicle. See Johnson Criteria.

Clutter: Objects in the background scene which interfere with the ability of an observer to acquire and distinguish targets.

Component: One of the sensors mounted on a hunter-killer platform. Each component may have its own sensor type, search sector, search azimuth, and Johnson level to which a target must be acquired before the component will take action. See hunter and killer.

Contrast (Target-to-Background Contrast): The relationship betweeen the radiance of a target and the background against which it is seen:

$$C = (C_t - C_b) / C_b \quad ,$$

where C is contrast, C_t is target radiance, and C_b is background radiance.

Contrast Transmittance: The ratio of apparent contrast to inherent contrast. The contrast reduction is caused by light scattered into the field-of-view of a sensor by the atmosphere between the sensor and the target.

$$T_c = C_a / C = 1 / [1 + S (1/T - 1)] ,$$

where T_c is contrast transmittance, C_a is the apparent contrast, S is the sky-to-ground ratio, and T is atmospheric transmittance over the path.

Critical Dimension: A target size parameter, usually height or the square root of the area of the target side facing the sensor, against which an imaging system's resolution is evaluated to determine target acquisition potential.

Day Sight or Visual Sight: Optics such as binoculars or day periscopes which only magnify the target image. In a day sight, the human eye is the image detector, limiting wavelength response to the 0.4-0.7 micron region.

Defense Mapping Agency (DMA): The government agency that produces the digitized terrain elevation tapes used for high resolution terrain modeling. Technical inquiries on acquiring and interpreting tapes may be directed to: Director, DMA Aerospace Center, ATTN: PPGD, 3200 South Second Street, St. Louis, MO 63118-3399. Phone: Autovon 693-4546 or commercial (314) 263-4546.

Detection: The ability to distinguish that an artifact within the field-of-view is of military interest. For thermal systems, detection is of two types: MDT or MRT. See Johnson Criteria.

Direct View Optics (DVO): See day sight.

Extinction: See Attenuation.

Forward Line of Troops (FLOT): An imaginary line constructed to demark the territory controlled by each side in a battle.

Field of Regard (FOR): An alternate term for Field of Search.

Field of Search (FOS): The area in which a sensor scans for targets. The area is defined by a center azimuth giving the direction of the search. The horizontal and vertical widths of the search area are given in degrees.

Field of View (FOV): The portion (angle) of the object scene which is included in the displayed imagery of an imaging system. Systems may operate in several FOV modes by changing the optics. NFOV, MFOV, and WFOV refer to narrow, medium, and wide FOV modes.

FLIR: Forward Looking Infrared, a type of sensor.

Hunter: A sensor mounted on a platform that is capable of detecting targets but is not capable of directing fire. See killer.

Hunter-Killer: An alternate term for platform.

Identification: The ability to discriminate the exact model of a target. For example, identification would allow the observer to distinguish a T-62 from a T-72 tank. See Johnson Criteria.

Image Intensifier (II): An imaging device using an electron tube that reproduces on a fluorescent surface an image of a radiation pattern focused on its photosensitive surface. Image intensifiers are used to produce an output image that is brighter to the eye than the original scene at some magnification. Image intensifiers respond to radiation in the 0.4-0.9 micron region and thus utilize some radiation which the eye is not able to detect in order to further improve their sensitivity.

Infrared (IR): Electromagnetic energy from 0.7 to 15 microns. Thermal imaging systems in this model use either the middle IR (3-5 microns) or far IR (8-12 microns) spectral regions.

Johnson Criteria: A set of four criteria used to define levels of target discrimination. The levels, in order of increasingly precise discrimination, are: detection, classification, recognition, and identification. Each level is defined by the minimum number of cycles from a standard bar type target pattern that must be distinguished for a target to be acquired at that level.

Killer: A sensor mounted on a platform that is capable of detecting targets and of directing fire on the targets. The hunter sensor may also detect targets and pass them to the killer to be fired upon. See hunter.

Line of Sight (LOS): A direct visual line from one point to another.

Macro-terrain: A term used in this model to refer to large terrain features, such as hills, buildings, or forests, which may block line of sight between two platforms. See micro-terrain.

Micro-terrain: A term used in this model to refer to small terrain features that may be used by a platform to mask itself from detection. Also called defilade.

MERADCOM: U.S. Army Mobility Equipment Research and Development Command.

Minimum Detectable Temperature Difference (MDT): A parameter used in the modeling of infrared imaging hardware performance. It is the minimum temperature difference between a square (or circular) target and the background necessary for an observer to perceive (but not necessarily resolve) a target source through the thermal imaging device. It is a function of target angular size and represents the threshold detection capability of the system.

Minimum Resolvable Contrast (MRC): A parameter used in modeling the performance of image intensifiers, day sights, and the eye. It is the apparent contrast required to resolve a target of a given spatial frequency, and is presented as a function of spatial frequency in units of cycles per milliradian.

Minimum Resolvable Temperature Difference (MRT): The central parameter in the modeling of infrared imaging hardware performance. It is both theoretically predictable and laboratory measurable. It is the minimum temperature difference required between a standard bar type target pattern (4-bar, 7:1 aspect ratio) at which a trained observer with normal vision can distinguish the bar pattern as a four bar pattern. The MRT is generally determined for a variety of different spatial frequency targets. It is a function of the minimum temperature difference in degrees Celsius versus spatial frequency in units of cycles per milliradian.

Molecular Extinction: The atmospheric attenuation of radiation due to absorption by or scattering from atmospheric molecules.

NV&EOC: The Night Vision and Electro-Optics Center.

Platform: The basic unit of the Hunter-Killer Model. A platform is a large piece of combat equipment such as a tank, APC, or truck. A platform may have sensors, may have a weapon, may be airborne or ground based, and may move about the battlefield. Also called a hunter-killer.

Precision Guided Munition (PGM): A munition which must be guided to its target. Some PGMs must be guided by the firing platform, such as a wire guided missle. Other PGMs may be guided by either the firing platform or by a different platform serving as a target designator, such as a missle that homes on a laser beam reflected off the target.

Probability of Kill (PK): See SSPK.

Radiance: The total radiant flux per unit solid angle per unit projected area which emanates from a surface.

Recognition: The ability to discriminate between two targets of similar type; i.e., recognition would allow the observer to distinguish between two types of tracked vehicles, such as APCs and tanks. See Johnson Criteria.

Remotely Piloted Vehicle (RPV): A small drone aircraft which carries a sensor and can be used to locate targets and to direct fire.

Resolvable Cycles: The number of cycles which the average observer using an imaging system can discriminate 50% of the time, under the set of conditions being considered.

Scattering: The removal of energy from a beam of radiation traversing a medium by reflection and refraction from particles of matter within the medium. Scattering particles (as within an atmospheric aerosol) have a different index of refraction than that of the medium. Scattering varies as a function of the ratio of particle diameter to the wavelength of the radiation.

Sector: Another term for the field of search for a component of a platform.

Single Shot Probability of Kill (SSPK): The probability of hitting and killing a target that is at a given range from the shooter. All SSPKs are given as a function of the type of weapon, the type of target, and the range.

Sky-to-Ground Ratio: The ratio of the radiance of the horizon sky to the inherent radiance of the background of a target, with reference to a given viewing geometry. The sky-to-ground ratio will vary in a complicated fashion with viewing angle, sun angle, and environmental and atmospheric conditions. See Contrast Transmittance.

Spatial Frequency: A term which describes the frequency of an evenly spaced bar type target pattern or a sinusoidal type target pattern. It is the number of cycles of the pattern which occurs in a given distance. The distance is usually expressed in millimeters or the more common angular distance of milliradians. The milliradian dimension is most commonly used for systems analysis becouse it projects the spatial frequency into object space.

Target Discrimination Level: The number of cycles that may be resolved across a target's critical dimension. The number of resolved cycles can be related to the probability of accomplishing a particalar task (e.g., target detection) by criteria such as described in Section 2.0.

Target Signature: See contrast and thermal signature.

Thermal Imaging: Pertaining to a class of devices that optically collect infrared radiation within a limited wavelength band (e.g., 3-5 or 8-12 microns) and convert the received energy into a "thermal image" of the scene which can be viewed by the human eye.

Thermal Signature or Delta T: The mean target-to-background temperature difference, referenced to a 23 degree Celsius background temperature.

Type Platform: A data structure used within the model to describe the characteristics common to all platforms of the same type. For example, the data set for the model may contain four M1 tank platforms, each with its individual search strategies and movement vectors. The model would then contain a single M1 tank type platform that would describe the common characteristics such as height, weapon type, delta T, and contrast.

Visibility or Visual Range: A measure of target detection range that depends only on the extinction of the atmosphere in the visual (0.4-0.7 micron) spectral region. Visibility (V) is defined as the range at which visual atmospheric transmittance is 0.02 and can be found from the visual extinction coefficient s_V by

$$V = 3.912 / s_V .$$

Visible Radiation (or Light): Electromagnetic radiation in the wavelength range of approximately 0.4-0.7 microns, the wavelengths to which the human eye is sensitive.

This page was intentionally left blank.