NAVAL POSTGRADUATE SCHOOL
Monterey, California

DTIC
S ELECTE D
JUN 01 1995
F

THESIS

A MODEL FOR ESTIMATING THE COLLATERAL DAMAGE RESULTING FROM AN INTERCEPTED THEATER/TACTICAL BALLISTIC MISSILE

by

Joseph M. Oliver

March 1995

Thesis Advisor: Donald P. Gaver

Approved for public release; distribution is unlimited.

19950531 011

REPORT DOCUMENTATION PAGE			Form Approved OMB Np. 0704
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washingon headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE March 1995	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE A MODEL FOR ESTIMATING THE COLLATERAL DAMAGE RESULTING FROM AN INTERCEPTED THEATER/TACTICAL BALLISTIC MISSILE			5. FUNDING NUMBERS
6. AUTHOR(S) Joseph M. Oliver			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10.SPONSORING/MONITORING AGENCY REPORT NUMBER
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

The ability to produce an accurate estimate of the expected amount of damage that will occur to a targeted region from the fragments generated by intercepting a Theater/Tactical Ballistic Missile (TBM) in flight is necessary for determining whether the TBM should or should not be intercepted. For a sea-based Theater Missile Defense (TMD) system capable of predicting the remaining portion of the TBM's trajectory along with its impact point from its current and past flight parameters, possessing this ability may prevent unnecessary intercepts, and thus unnecessary damage. This thesis proposes a simplistic simulation model that produces expected damage estimates for various intercept ranges. Damage is evaluated in three areas: number of fragments to impact the target region, total mass to impact the target region, and total kinetic energy to impact the target region. The values generated by this model can aid in the decision of whether to intercept the TBM or not by comparing them against the expected damage values caused by the unintercepted TBM which are calculated by the sea-based TMD system.

14. SUBJECT TERMS Theater/Tactical Ballistic Missile, Theater Missile Defense.			15. NUMBER OF PAGES 149
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

Approved for public release; distribution is unlimited.

A MODEL FOR ESTIMATING THE COLLATERAL DAMAGE RESULTING FROM AN INTERCEPTED THEATER/TACTICAL BALLISTIC MISSILE

by

Joseph M. Oliver
Lieutenant, United States Navy
B.S., Villanova University, 1989

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1995

Author: Joseph M. Oliver
Joseph M. Oliver

Approved by: Donald P. Gaver
Donald P. Gaver, Thesis Advisor

Arnold Buss
Arnold Buss, Second Reader

P. Purdue
Peter Purdue, Chairman
Department of Operations Research

Accesion For		
NTIS CRA&I		☑
DTIC TAB		☐
Unannounced		☐
Justification		
By		
Distribution /		
Availability Codes		
Dist	Avail and / or Special	
A-1		

ABSTRACT

The ability to produce an accurate estimate of the expected amount of damage that will occur to a targeted region from the fragments generated by intercepting a Theater/ Tactical Ballistic Missile (TBM) in flight is necessary for determining whether the TBM should or should not be intercepted. For a sea-based Theater Missile Defense (TMD) system that is capable of predicting the remaining portion of the TBM's trajectory along with its impact point from its current and past flight parameters, possessing this ability may prevent unnecessary intercepts, and thus unnecessary damage. This thesis proposes a simplistic simulation model that produces expected damage estimates for various intercept ranges. Damage is evaluated in three areas: number of fragments to impact the target region, total mass to impact the target region, and total kinetic energy to impact the target region. The values generated by this model can aid in the decision of whether to intercept the TBM or not by comparing them against the expected damage values caused by the unintercepted TBM which are calculated by the sea-based TMD system.

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without verification is at the risk of the user.

TABLE OF CONTENTS

ACKNOWLEDGMENTS

The author wishes to thank many of the faculty and staff of the Naval Postgraduate School who showed an interest in this thesis topic and helped point me in the right direction.

I would like to thank Professor Dan Holland of the Physics Department for his assistance with the mechanics behind the model. I would also like to thank Kenn Gerchman from the Operations Analysis Department for his assistance with the UNIX operating system. I would like to extend special thanks to Dr. Bruce Fowler of U.S. Army Missile Command for his timely and constructive review of this thesis.

Lastly, I would like to thank my thesis advisor, Professor Donald P. Gaver, and second reader, Professor Arnold Buss, for their patience and their significant recommendations throughout the entire thesis process. They dedicated a great deal of time to this study.

EXECUTIVE SUMMARY

PROBLEM: A model for estimating the expected collateral damage to a target region resulting from the intercept of a Theater/Tactical Ballistic Missile (TBM) is necessary for the decision maker to determine if intercepting the TBM is the proper course of action and is the focus of this thesis.

With the increased threat of theater/tactical ballistic missile attacks against defenseless regions, and the propensity of some nations to use ballistic missiles as a bullying tactic, theater missile defense (TMD) has become a high-priority issue for the Department of Defense. Presently, TBM's pose a great threat to U.S. military operations because almost every conceivable future conflict involving U.S. military deployments would involve the risk of TBM attacks against U.S. forces. The only existing deployable TMD system which intercepts a TBM in flight, Patriot, an Army air defense system adapted for the TMD role, has suffered severe criticism for its performance in Desert Storm. At first, Patriot was praised as an extremely effective interceptor of SCUD missiles launched from Iraq at Israel and Saudi Arabia. Unfortunately, post-war analysis showed that the Patriot missile was much less effective than was initially claimed. Patriot is now being modified to enhance its capabilities against TBM's. Another land-based TMD system under development is the new Theater High Altitude Area Defense system (THAAD). The integration of THAAD and an improved Patriot promise to provide outstanding defense against future ballistic missiles.

In addition to these systems, the Ballistic Missile Defense Organization (BMDO) has directed the Navy to adopt this two-tiered approach of exo-atmospheric and endo-atmospheric defense and develop its own TMD systems to be integrated with the Army TMD systems in the future. The weapon chosen for the lower-tier, or endo-atmospheric, role is SM-2 Block IV A surface-to-air missile. One concern with the future testing of this missile is the unknown magnitude of the collateral damage that results from a TBM intercept. Therefore, a model is proposed for estimating the expected damage to a target region and its surrounding areas.

SOLUTION METHODOLOGY: This thesis addresses the question of when to intercept a TBM and when not to intercept it. The model proposed is a simplistic analytical attempt at estimating the distribution of the expected damage that occurs to a target region and its surrounding areas from intercepting a TBM at predetermined points on its flight path. The estimate can then be compared against the amount of damage that would have occurred if the TBM were allowed to continue without intercept, assuming the TMD system in question has this capability.

Utilizing MODSIM, a simulation, which replicated the behavior of the fragments generated after an intercept, was developed. Because of the lack of available data concerning the true fragmentation of a TBM upon intercept, many simplifying assumptions were made. In order to account for the uncertainty in the TBM's impact point, if not intercepted, two parameters of the TBM's initial conditions were allowed to vary in each run: vertical launch angle and off-target launch angle. Replications of the simulation were conducted for various intercept points and pairs of launch angles so as to generate data on the number of hits, total mass and total kinetic energy impacting the target region. The distributions of the number of fragment hits, the total mass impacting the target region and the total kinetic energy impacting the target region are examined to determine expected values.

SOLUTION: The modeling and simulation results presented are speculative and exploratory. They illustrate the effects of various physical influences upon missile collateral damage to a target in a preliminary way. The conclusion drawn from this thesis is that knowing the expected damage to occur to a target region if a TBM is intercepted is helpful in making a decision whether to intercept it or not. This model is a simple analytical attempt to produce a tentative estimate. At this time, very little is known concerning the dispersion of fragments generated by an intercept. Because of this lack of information, a crude estimate is the only estimate available. When the assumptions made in this thesis are determined with certainty, a more detailed version of the model, replicating the exact flight paths and impact points, can be developed.

I. INTRODUCTION

With the increased threat of theater/tactical ballistic missile (TBM) attacks against defenseless regions, and the propensity of some nations to use ballistic missiles as a bullying tactic, theater missile defense (TMD) has become a high-priority issue for the Department of Defense. Presently, TBM's pose a great threat to U.S. military operations because no truly effective ballistic missile defense systems exist. Moreover, almost every possible future conflict involving U.S. military deployments would also involve the risk of TBM attacks against U.S. forces. The most familiar TBM, encountered extensively during the Gulf War, is the SCUD, Variant B. Many countries have these missiles in their arsenals including such unstable nations as: North Korea, Iraq, Vietnam, Yemen, Libya, Afghanistan, and Syria [Ref. 1].

The only existing TMD system which intercepts a TBM in flight, the Patriot, an Army air defense system adapted for the TMD role, has suffered severe criticism for its performance in Desert Storm. At first, Patriot was praised as an extremely effective interceptor of SCUD missiles launched from Iraq at Israel and Saudi Arabia. Unfortunately, post-war analysis showed that the Patriot missile was much less effective than was initially claimed. In fact, the majority of SCUDs thought to be intercepted had spontaneously broken into segments upon re-entry as a result of faulty manufacturing. Patriot is now being modified to enhance its capabilities against TBM's. Another land-based TMD system under development is the new Theater High Altitude Area Defense system (THAAD). The integration of THAAD and an improved Patriot promise to provide outstanding defense against future ballistic missiles. However, THAAD development is being decelerated because of budget cuts. The drawback of both Patriot and THAAD, when the latter is completed, is the enormous lead time required to deploy and set up these systems. Future TBM attacks on areas of U.S. interest may not be known weeks in advance; time to deploy these defense systems may not be available. Also, permission must be granted from foreign leaders to deploy land-based TMD systems. The leaders may be hesitant even though these systems are there to save their territory because deployment of such systems on their land may escalate hostilities. In addition to the deployment lead time is the enormous expense of airlifting the systems to the threat areas. In summary, such drawbacks provide strong motivation for the development of a *sea-based ballistic missile defense system*.

As mentioned previously, plans and initial steps necessary for TBM attacks may not be known long enough (e.g., weeks) in advance of attack to allow for deployment of Patriot batteries. A timely response option is the use of a mobile sea-based ballistic missile defense system. The Navy is in the process of developing a version of this system by upgrading the Aegis Weapon System and the SM-2 Block IV standard missile in order to effectively detect, track, engage, and intercept TBM's. The new standard missile is to be an improvement of the existing vertical-launch Terrier/standard surface-to-air missile. The latter requires modifications of the warhead, fuze, and seeker to improve its capability to defend against TBM's. The new version would build upon already-developed engineering and design efforts used on Patriot, but there are drawbacks to using this technology [Ref. 1]. One of these is the explosive warhead. This warhead may cause total destruction with a near-miss intercept, or it may not. If the near-miss intercept does not totally destroy the TBM, debris from the TBM could impact the intended target or surrounding areas.

Collateral damage is the term for non-target damage caused by the debris. For example, if a TBM is targeted to impact the center of a coastal city and U.S. forces are conducting an amphibious assault, intercept of the TBM might stop it from directly impacting its target, but the fragments generated by the intercept might fall on the U.S. forces and interfere with the assault operation. When contemplating an intercept, the pattern of the debris generated from the intercept must be estimated in order to determine the amount of damage that will occur if the ballistic missile is intercepted and fragments continue on course. According to Israeli reports after Desert Storm, the 11 SCUDS intercepted by Patriot batteries caused more damage than did the 13 SCUDS that directly hit targets within Israel. This raises an interesting question: with prior knowledge as to the amount of damage, both primary and collateral, that will occur from intercepting a TBM, is intercepting the TBM always the best strategy? If the TBM is allowed to continue on course without intercept, it is possible that the amount of damage resulting from the single impact point may be less than that of a shower of fragments impacting the target area. Of course, there is also a chance that the TBM will miss the target area completely. Because the dispersion of fragments resulting from an intercept and the size of each fragment are unknown, an accurate estimate of the debris pattern is almost impossible. This dispersion uncertainty may be caused by numerous aspects including angle of impact by interceptor, angle of descent of ballistic missile, impact point on ballistic missile, distance between interceptor and ballistic missile at intercept.

Purpose of this Thesis: Where, and If, to Intercept?

In order to address the question as to when to intercept a ballistic missile and when not to intercept it, a simulation model is developed to estimate the distribution of the damage that occurs from intercepting a TBM at predetermined points on its trajectory. This damage is then compared to the amount of damage that would have occurred if the TBM were allowed to continue without intercept. Development of the model requires a number of simplifying assumptions to be made. The first, and possibly most credible, set of assumptions deals with the characteristics of the TBM, including launch acceleration, motor burn time, missile length, missile diameter and missile mass. The second, more speculative, set of assumptions concerns the generation and motion of fragments, including number of fragments, mass of each fragment, cross-sectional area of each fragment, and the velocity of each fragment after intercept. These characteristics are generated from various probability distributions judged to be appropriate. The final assumption is target region designation. For this study the target region is a coastal region that is 2 km X 2 km. This region is then broken up into 16 blocks that are 0.5 km X 0.5 km each. Actual targets that are subregions of this region can be studied for damage.

The preceding assumptions remain the same for each run of the model. However, two parameters are allowed to vary in each run: vertical launch angle and off-target launch angle. For each pair of angles, while holding all other parameters constant, a different TBM flight path is generated. From these flight paths an impact point could be calculated. Although, having precise knowledge of these two angles is not realistic, determining the impact point of a TBM from its flight path, conditional on the angles, is realistic. For simulation purposes the two angles are taken to be given and known in order to calculate the impact point. Realistic expected impact damage can be estimated by averaging over any angle distribution of interest.

The model is intended to represent a worst-case situation: the intercept results in a near-miss warhead explosion that creates missile fragments which continue traveling towards the target area with minimal loss of mass and minimal change in velocity. The TBM's flight path is modeled by using suitable equations of motion in three dimensions. At pre-determined points on the flight path, the hypothetical intercept points, the missile breaks up into a random number of fragments resulting from the near-miss intercept. Each fragment has the velocity of the TBM at intercept plus a random "kick" in three dimensions to simulate breaking apart. Each of the fragments continues towards the target region on a path determined by its mass and cross-sectional area. This model treats the cross-sectional areas of the fragments as functions of their respective masses, i.e., the larger the mass the larger the cross-sectional area.

The objective of this modeling project is to estimate the expected damage that occurs when the TBM is intercepted at various distances from the target area, given the vertical and off-angle launch angles. Certain intercept points generate fragments which impact the target area. The amount of damage that occurs from a fragment impact is estimated using a damage function that takes into account fragment mass and velocity. The damage generated from different intercept points is then compared to the damage generated by letting the missile through the defenses with hope of a miss. Replications of this simulation are conducted for various intercept points and pairs of launch angles so as to generate data on the numbers of hits, total mass and total kinetic energy impacting the target area. The distributions of the number of fragment hits, the total mass impacting the target area and the total kinetic energy impacting the target area will be examined to determine expected values and summaries of the effect of intercepting missiles at various locations along their trajectories. The ballistic missile modeled in these simulations are based on the characteristics of the SCUD variants used in Desert Storm, i.e., they will have a fairly large circular error probable (CEP) which is represented by the varying angles.

The modeling and simulation results presented are speculative and exploratory. They illustrate the effects of various physical influences upon missile collateral damage to a target in a preliminary way. The model equations and software, which are provided herein, can be modified or extended to test sensitivity to specific parameter values. With slightly more effort the actual physical relationships that have been used can, if considered necessary, be modified in more reasonable and defensible ways.

II. NATURE OF THE PROBLEM

A. PROBLEM DESCRIPTION

As demonstrated in the Gulf War, Iraq used it's Al Hussein TBMs, i.e. modified SCUD B's, against civilian targets, most of which were populated cities. The circular error probable (CEP) of this particular TBM is fairly large, being about 1000 m, allowing a chance of missing the target altogether and landing in some insignificant area, e.g., the sea. Actually, this happened quite often during the Gulf War. Of the 88 TBMs launched by Iraq towards Israel and Saudi Arabia, 35 missed their targets completely, landing either in the sea or in the desert. Future sea-based TMD systems will be able to estimate, with a level of certainty, the impact point of a TBM by calculating the remaining portion of its trajectory according to the TBM's current and past parameters. However, these systems do not rely solely on the estimate in determining what action to take, since the estimate is not exact. If the impact point is judged likely to be close to the target area, one of two courses of action must be taken: intercept the missile and risk collateral damage by the missile fragments, or do not intercept the missile and hope nothing significant is located near the impact point. Therefore, the problem to be addressed in this section is to determine whether a sea-based TMD system, assigned to protect a targeted region, e.g., a coastal city, *should intercept* a TBM in flight, creating missile fragments that may still impact the target area, or *should refrain from interception*, allowing the TBM to continue towards the intended target area, having assessed that it will miss.

To address this problem, a computer simulation utilizing MODSIM, a simulation software system, is designed to determine the amount of damage that occurs from intercepting a TBM in flight. In this simulation a TBM modeled after the Al Hussein, but with generic numbers substituted for some of its characteristics to preserve the unclassified status of this project, is launched toward a predefined target area. Parameters of the TBM are first chosen to allow an impact point at the exact center of the target area. Points along the TBM's flight path are selected as intercept points, at which the missile separates into a random number of fragments that continue towards the target. The final position, or impact point, of each of the fragments generated is examined to determine if the fragment landed inside the targeted area, and where therein. Summing the results for each fragment totals the amount of damage to occur for a given intercept point. Conducting numerous replications of the model for each intercept point yields the average damage to occur to the target area. These results are further analyzed using statistical techniques to determine the expected damage to occur for various intercept points.

Since this model is deterministic, the unintercepted TBM, under the assumption of a flat, non-rotating earth, would have the same impact point, i.e., the center of the target area, for each simulation run unless one or more parameters are allowed to vary. By changing parameters on each simulation run, the TBM's impact point, if allowed to continue towards the target without intercept, also changes. The parameters chosen to vary in this simulation are the vertical and off-target launch angles as depicted in Figure 1. By introducing a small degree of error in either angle, the TBM impacts vary from the aim point. As mentioned previously, a sea-based TMD system will be able to estimate, fairly accurately, this final hypothetical impact point of the TBM by constructing the remaining trajectory using its current and past parameters: course, speed, angle of descent, altitude, and others. Hence, a decision can be made either to intercept the TBM, and where along its path, or to allow it to continue on course by comparing the damage caused by a TBM impact at the point estimated by the sea-based TMD system against the calculated damage caused by fragments impacting the target area.

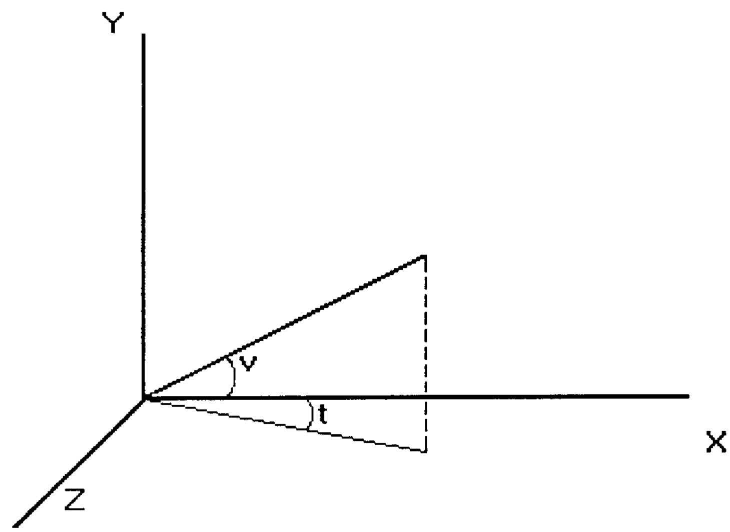


Figure 1. 3-Dimensional Plot of the TBM Launch Angles

t = off-target launch angle

v = vertical launch angle

B. PROBLEM ASSUMPTIONS

To preserve the unclassified status of this thesis and to allow the expressions utilized in developing the TBM trajectories to be closed form, the following assumptions are made:

- The sea-based TMD system is stationed somewhere between the launch point and target where all potential intercept points utilizing an SM-2 Block IV A Standard missile are feasible. The position of the ship will not coincide with the debris pattern.

- Initial detection of the TBM is via space-based assets which provide a cue to the sea-based TMD weapons system. The sea-based TMD system picks up the TBM track as soon as it enters the detection envelope, giving it enough time to calculate an impact point. Error in that estimate will realistically occur, but is not considered in what follows.

- The attack on the target consists of only one TBM.

- The TBM experiences constant acceleration and straight-line motion during boost phase with air resistance having no effect.

- The TBM follows a normal ballistic path, executing no maneuvers to avoid an intercepting missile.

- The TBM remains intact throughout the flight and the mass remains essentially constant during boost phase even though fuel is being expended. At the completion of boost phase the TBM's mass is decreased by one seventh of its original mass to compensate for the spent fuel.

- The intercept simulates a near-miss explosion of the SM-2's warhead which separates the TBM into several fragments, the number and sizes are simulated from a combination of plausible probability distributions; these latter can be modified as desired by an analyst.

- There is minimal loss of mass and minimal change in velocity resulting from the intercept.

C. MODEL OBJECTIVE

If it is estimated that the TBM will impact in the vicinity of the target, intercepting it towards the end of its flight path at low altitude may generate fragments that still impact the target region and cause collateral damage. Intercepting it earlier in its flight path obviously presents a smaller chance of collateral damage. With this knowledge, the ideal intercept is the farthest distance possible from the target where intercept is feasible. However, at this distance the impact point of the TBM estimated by the sea-based TMD system is fairly inaccurate, and an unnecessary intercept may occur, i.e., the TBM would have fallen short of its target and landed in the sea, requiring no intercept. Before planning an engagement, knowing the damage that results from different intercept points allows the sea-based TMD system to decide whether an intercept is necessary. Therefore, the objective of this model is, for particular intercept points, to *estimate the expected values for the number of fragment hits, the amount of mass to impact the subregions of the general target region, and the total kinetic energy to impact subregions of the general target region* in order to decide whether the damage sustained from the fragments will be greater than that of the TBM itself. The above measures of TBM effect are actually tabulated for subregions of the general target region. Note that the simulation allows evaluation of measures of target (sub)region effects other than *expected* values: it is, for example, possible to estimate the *probability* that the number of fragments that impact in a particular subregion exceeds a prescribed level, or the probability that the total kinetic energy delivered to a particular subregion exceeds a given level. This capability of the model is appropriate when hardened military targets are in the area, and their vulnerability to damage is low unless a specific threshold is exceeded.

III. MODEL

The purpose of this thesis is to develop a simulation model that produces a tentative estimate of the expected damage that occurs within a designated target region caused by the fragments generated from intercepting a TBM. To accomplish this task the model may be considered as having two parts. The first part represents launching a TBM towards a designated target region located some known distance from a selected launch point. The TBM is tracked from launch through its trajectory until it reaches the intercept point chosen by the user. The effect of tracking the TBM is simulated by formulating and solving three-dimensional equations of motion, utilizing the TBM's mass and cross-sectional area and physical laws.

The second part of the model begins with the TBM intercept, which is treated as a near-miss intercept that only breaks the TBM into pieces rather than destroying it. The reason behind this is to have the model represent a worst-case situation. If minimal mass and velocity are lost as a result of an intercept there stands a greater possibility of the fragments reaching the target region. At intercept the TBM breaks up into a random number of fragments. Each fragment generated continues towards the target region on its own flight path. These fragment flight paths are calculated using the same three-dimensional equations of motion as describe the TBM's flight path, but this time depending upon each fragment's mass and cross-sectional area. Of concern in this model is the extent of damage to the target region caused by these fragments. Therefore, the model determines which fragments impact the designated target region by comparing the final positions of each fragment to the target location. If it is determined that a fragment has impacted the target region, its mass and kinetic energy are recorded and subsequently totaled with the masses and kinetic energies of other impacting fragments. However, if it is determined that a fragment does not impact the target region, it is considered a miss, and the mass and kinetic energy are recorded for that region surrounding the target where the miss occurred.

Each run of the present model is analyzed by statistical techniques to estimate the expected number of fragments to impact various subregions of the target, and the expected total mass and expected total kinetic energy to impact these subregions. Conducting 2000 replications of the model for each intercept point enabled Monte Carlo analysis to be implemented to produce a fairly accurate estimate of these expected values. This many replications were required to keep the standard error of the estimate at a desired low level. The following section describes the model in depth.

A. SIMULATION DESCRIPTION

1. Initial Conditions

As mentioned previously, the TBM utilized for this thesis is being modeled after Iraq's version of a modified SCUD, the Al Hussein. For classification purposes, only four actual characteristics of the Al Hussein are used in this simulation. The remaining characteristics required for the simulation have been calculated based on assumptions made concerning the distance to the target region. Another assumption concerning the flight of the TBM was that the calculations were based on a flat, non-rotating earth. For this thesis, the only portion of the TBM's trajectory of interest is immediately pre-intercept to impact. Therefore, the manner in which the TBM's flight path is calculated is artificially simplistic and thus is performed as a means of obtaining the trajectory portion of interest. The characteristics of the TBM modeled in this thesis, along with the calculations necessary to compute the simplistic TBM flight path are contained in Appendix A.

2. Intercept and Fragment Generation

In order to intercept a TBM, an actual interceptor of some kind is required. However, since the objective of this model is to determine the extent of damage to a target region caused by TBM fragments, *given* that an intercept has occurred, modeling of the interceptor is omitted. Therefore, to simulate the effect of intercept, the TBM breaks (spontaneously) apart into a random number of fragments, each experiencing a minimal change in velocity from that of the TBM because most of the energy created by the intercept is consumed fragmenting the TBM. Also, the accumulated mass of all the fragments generated is taken to be close to that of the TBM. Lacking any data to validate, these assumptions are as good as possible. Accepting these two conditions, minimal changes in velocity and mass, it is plausible that the intercept represents a worst-case situation.

Determining the exact number of fragments, and the size of each fragment, that result from intercepting a TBM is impossible because of numerous factors including angle of impact by interceptor, angle of descent of ballistic missile, impact point on ballistic missile, distance between interceptor and ballistic missile at intercept, and strength of TBM materials. Therefore, *estimates* of the random number of fragments and of the individual fragment characteristics, i.e., mass, cross-sectional area, and initial velocity, are introduced to complete the model.

The actual realized numbers of fragments and the individual fragment masses that occur in one simulation replication are developed using a two-step operation. In the first step, a random number is drawn from the Poisson probability distribution. The mean number of fragments desired for this model, which is requested from the user at the start of the simulation, is used in generating this Poisson random number. This number is then divided into the TBM's mass to get the average mass of a fragment. The second step involves a loop which generates fragments one at a time. This loop is dependent solely on the mass of the fragments and continues until the accumulated mass of the fragments is greater than that of the TBM. Each time through the loop a fragment mass is generated by multiplying the average fragment mass produced in the first step by an Exponential random variable with a mean of one. This process simulates the variability between fragment masses. Each fragment mass is then added to the accumulated total. So long as the total is less than the TBM's mass, a fragment is generated. When the accumulated total becomes greater than the TBM's mass, fragment generation ceases and the number of fragments is determined. Of course modifications of the above procedure can be made in the simulation code: both Poisson and Exponential distributions are provisional illustrative choices.

The remaining characteristics, cross-sectional area and initial velocity, of each fragment are also calculated during each run through the loop. Three assumptions were made in order to specify the cross-sectional area of a fragment. First, since the exact shape of each fragment generated is unknown, this model proposes that "the" cross-sectional area is a function of the fragment's mass. Second, it assumes that each fragment, at the instant of intercept, takes on the (approximate) shape of a sphere. The last assumption is that the mass density of the fragment equals that of the TBM. With these assumptions, each fragment's cross-sectional area is calculated using simple geometric equations:

$$Volume = \frac{Mass}{Density} \tag{18}$$

$$Area = \pi \cdot radius^2 \tag{19}$$

Once again, because of the lack of knowledge concerning the way in which fragments are generated after an intercept of a TBM, a presumption as to how the model should handle fragment velocities was necessary. The initial velocity of each fragment is fairly easy to calculate, assuming the fragments are dispersed in a Gaussian manner, i.e., the TBM breaks up in a conical fashion, and that there is negligible loss of velocity. To accomplish this Gaussian dispersion, a random "kick" was applied to the three components of the TBM's velocity at intercept. The amount of this "kick" was determined by a Normal random variable with a mean of zero and standard deviation of ten meters/second. This allows each fragment to have a different initial velocity and, when combined with different masses and cross-sectional areas, a different flight path.

3. Fragment Flight Paths

The fragment flight paths are calculated in the same manner as is that of the TBM. A loop is initiated at intercept for each fragment with its initial conditions estimated under the above assumption. At each pass through the loop the fragment's velocity is calculated using equations (12) - (14) from Appendix A. The fragments experience a significant reduction in their velocities. This change is the result of greater air resistance being applied to the fragments mainly because of the reduction in their respective masses. Fragments with larger masses experience smaller reductions in velocity than do those with smaller masses. The positions of the fragments along their paths are then updated using equations (15) - (17) from Appendix A. Prior to completing each pass through the loop an extra step, not used in determining the flight path of the TBM, is implemented. The extra step concerns the cross-sectional area. In calculating the TBM's flight path, which in this model is stable, the cross-sectional area remained constant. However, it is believed that the flight of the fragment is *not* stable, and thus the cross-sectional area of the fragment does not remain constant. To account for this instability, a new cross-sectional area is calculated during each pass of the loop. This is accomplished by generating a random number using a Normal distribution in which the original cross-sectional area acts as the mean, and the standard deviation is 0.05 meters^2. One might speculate that the drag coefficient, C, changes along with the changing cross-sectional area. However, that case is not address in this thesis. One thing to remember is that this procedure is simply a means to simulate the unstable flight, and does not represent the actual way in which a fragment travels. Upon completion of a fragment's flight path the loop is exited and the model conducts damage assessment by locating the fragment's hit position within a target subregion.

4. Target Designation

Defining a specific target in the model, e.g., a particular populated city in the middle east, is too restrictive for purposes of this thesis. The objective is to estimate the damage to some designated target region caused by fragments generated from an intercepted TBM. The model must be adaptable for different targets. To accomplish this, however, one restriction is necessary. The dimensions of the target region must be rectangular. This rectangular region is then divided into equal-sized blocks, or subregions, to allow finer detail in estimating target damage. The number of blocks depends on the desired number of rows and columns requested from the user at the start of the simulation. The generic target region used in this example is depicted in Figure 2. The exact center of the rectangle is designated as the aim point. This rectangle then acts as a damage template which is placed over the target region with the bottom of the rectangle, blocks 13 through 16, being perpendicular to the TBM flight path. Figure 3 displays a target region composed of blocks 6, 7, 10 and 11. If the intended target is not rectangular in shape or is comprised of a number of smaller targets, i.e., runways at a coastal airport or piers at a naval port, the size of the rectangle is increased to encompass the total target area and only the blocks containing the actual targets are considered areas of interest. Figure 4 depicts piers at a naval port composed of blocks 5, 6, 7, 9, 11, 13, and 15. The remaining blocks are considered areas of insignificance. However, a target that is located within a template block is required to occupy the entire block, otherwise the damage

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Figure 2. Layout of generic target region template

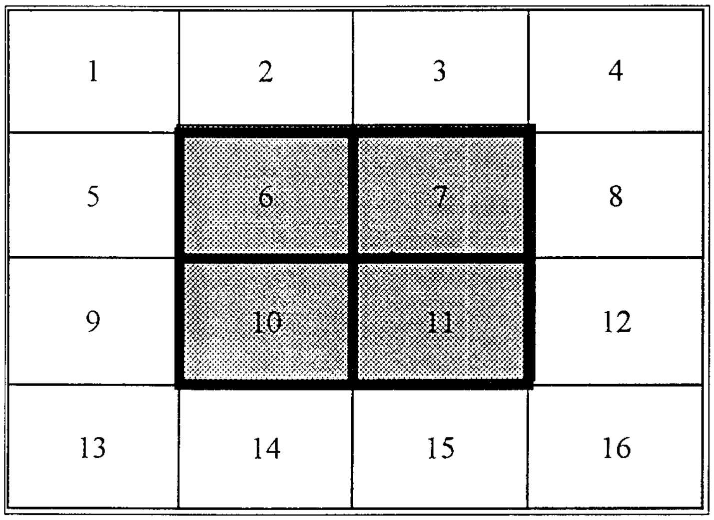


Figure 3. Target region comprised of blocks 6, 7, 10, and 11.

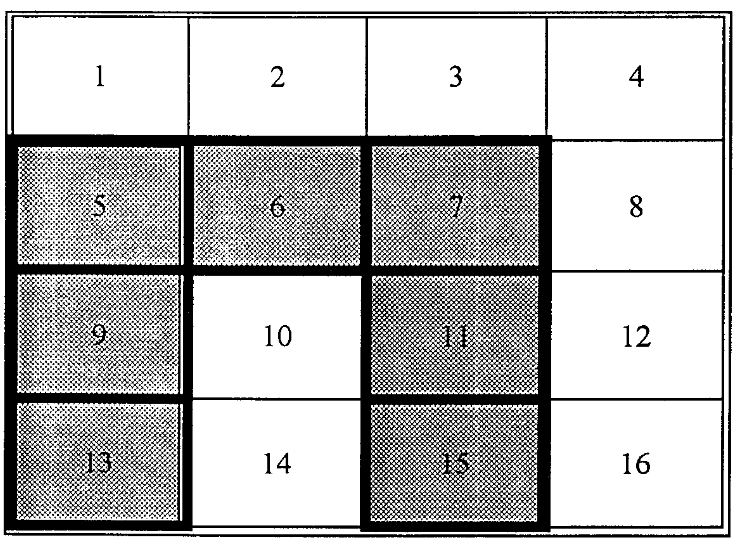


Figure 4. Layout of target region consisting of piers at a naval port.

assessment is inaccurate because the damage suffered is not uniform throughout a block. This encompassing feature of the target rectangle allows the target to be of any size. However, smaller blocks produce more accurate damage estimations because of the finer detail. Assuming that all targets are rectangle is fairly unrealistic, but using this assumption for damage assessment only, allows this kind of model to be used for just about every target imaginable that fits within the outlined target region.

5. Damage Assessment

This model assesses damage by three measures: *the mean number of fragments* to impact the target subregions, *the mean total mass* to impact the target subregions, and *the mean total kinetic energy* to impact the target subregions. The final positions of each fragment generated from a single intercept are compared against the location of the designated target region through a series of if-then statements to determine the extent of damage. A realistic way of determining damage with this damage template is to assume that the entire block is a target, i.e., there are no insignificant portions of the target block. Since this model deals with a generic target region, a generalization is made about the damage that occurs to a target block. If a target occupies an entire block then the damage that occurs to that block is treated as one amount even though fragments may impact at several different points within the block. If it is determined that the fragment impacted the target region, the estimated damage totals of the target block which contains that fragment's final position are updated. The counter for the number of fragments to impact is incremented, the accumulator for the total mass damage has the mass of the fragment added to it, and the accumulator for the total kinetic energy damage has the kinetic energy of the fragment added to it. The kinetic energy associated with fragment i is calculated by the following:

$$KE_i = \frac{1}{2} m_i v_i^2 \qquad (20)$$

where

v_i = velocity of fragment i at impact

m_i = mass of fragment i at impact

In order to produce mean damage values with small standard errors, at least 2000 replications of the model are necessary. At the completion of each replication the totals for each block are recorded to be used in calculating the means and the standard errors of the means using the following formulas:

$$\overline{D}_{ib} = \frac{1}{R}\sum_{j=1}^{R} D_{ib}(j) \qquad (21)$$

$$s^2 = \frac{1}{R-1}\sum_{j=1}^{R}(D_{ib}(j) - \overline{D}_{ib})^2 \qquad (22)$$

$$SE = \sqrt{\frac{s^2}{R}} \qquad (23)$$

where

$D_{ib}(j)$ = type i damage value for block b in the jth simulated replication

$\overline{D}_{ib}$ = mean type i damage value for block b

R = number of replications

s^2 = sample variance of damage value

SE = standard error of the mean

In addition to estimating the extent of damage to the target region, the model also calculates the amount of damage that occurs to the target's surrounding areas. This thesis considers these areas, "miss" areas. However, analysis is conducted in order to know the location of the remaining fragment damage in case less significant targets are located in these areas. Figure 5 displays these areas. Since the launch point could be from any direction, assigning the surrounding areas with a direction, i.e., north of the target, is not implemented. Instead, the model determines if the damage is past or over the target, or is short of it, or right or left of it.

Past & Left	Past	Past & Right
Left	TARGET	Right
Short & Left	Short	Short & Right

Figure 5. Plot of surrounding areas to the target region.

6. Structure of a Test

Many assumptions were made in this model concerning the fragmentation of a TBM upon interception. The reason for this is the lack of available data on both the true fragmentation and the characteristics of the resulting fragments. One possible way of validating this model is to structure a test that replicates an intercepted TBM and the resulting fragments. After completion of such a test, a distribution fitted to the fragment impact points determines if the assumptions made were correct leading to a valid model. An example of such a test, on a small and inexpensive scale, could use 2.75" rockets with ball bearings and a radio-controlled burster warhead as test vehicles. At predetermined points along the rocket's trajectory the radio-controlled burster is detonated releasing the ball bearings to travel their own paths towards the target region, or in this case the witness area. This area is covered in saran wrap enabling exact determination of the ball bearing impact points. The positions of these impact points are then fitted to a probability distribution to see if the model assumptions are correct. This is just one example of a possible test that can be used to validate the proposed model in this thesis since exact data is not available.

B. DESIGN OF A SIMULATION

A series of runs are conducted to estimate the damage that occurs to subregions within a designated target region from fragments generated by the intercept of a TBM as the two launch angles, vertical and off-target, are varied slightly to account for the different impact points associated with the TBM. To increase the coverage of this model and examine other aspects of the damage estimates, three different intercept ranges are simulated. Varying the intercept range allows an understanding of the need to intercept the TBM as soon as possible, thereby causing less damage to the target area. The MODSIM source code for the simulation is contained in Appendix C.

IV. ILLUSTRATIVE EXAMPLE

To aid in the comprehension of this model an illustrative example is provided. The TBM is launched towards a target located 560 km downrange, The initial conditions for the TBM described earlier are used: vertical launch angle of 30.0 degrees, off-target launch angle of 0.0 degrees, launch acceleration of 0.023875 meters/seconds^2 , cross-sectional area of 0.6082 meters^2 , boost phase time interval of 90 seconds, and missile mass of 6000 kg after boost phase. These parameters achieve a direct hit of the aim point if the TBM is not intercepted. The model output is highly sensitive to changes in these numbers.

The additional information necessary to conduct this simulation are the intercept range measured in kilometers downrange from the aim point, an expected number of fragments generated, and the length and width of the target measured in kilometers. For this example, the intercept ranges of interest from the aim point are 2, 3, and 4 kilometers, the expected number of fragments is 5 [Ref. 8], and 2 km for the length and width of the target, corresponding to a square target area of 4 km^2. The target region is divided into 16 equal blocks, or square subregions, each of area 0.25 km^2 . Table 1 lists the estimates of the expected number of fragments to impact each block, the expected total mass to impact each block and the expected kinetic energy to impact each block, along with the standard errors associated with each of the estimates for an intercept of 2 km. Plots of these results are displayed in Figures 6 through 11. Table 2 show the results for an intercept of 3 km, and the plots of these results are shown in Figures 12 through 17. Table 3 shows the results for an intercept of 4 km, and the plots of these results are displayed in Figures 18 through 23. The remaining vertical launch angles examined in this thesis are the set (29.8° , 29.9°, 30.1°, 30.2° , 30.3°). The remaining off-target launch angles are the set (0.025°, 0.05°, 0.075°, 0.10°, 0.15°). The off-target launch angles represent either side of the target line. The model was run for the right side of the target line. To get the data for angles to the left just transpose the data to the corresponding blocks. All possible combinations of these angles were analyzed at the three intercepts and the results are contained in Appendix B.

Block Number	Expected Value	Standard Error
6-Frag	0.067	0.004
6-Mass	267.819 kg	16.199 kg
6-K E	9.036 J	0.604 J
7-Frag	0.067	0.004
7-Mass	267.819 kg	16.199 kg
7-K E	9.036 J	0.604 J
10-Frag	0.67	0.012
10-Mass	1258.149 kg	22.499 kg
10-K E	18.81 J	0.46 J
11-Frag	0.67	0.012
11-Mass	1258.149 kg	22.499 kg
11-K E	18.81 J	0.46 J
14-Frag	0.29	0.009
14-Mass	274.421 kg	8.3 kg
14-K E	1.624 J	0.05 J
15-Frag	0.29	0.009
15-Mass	274.421 kg	8.3 kg
15-K E	1.624 J	0.05 J
Short-Frag	2.574	0.054
Short-Mass	986.021 kg	21.472 kg
Short-K E	546.039 J	12.338 J

Table 1. List of estimates of expected values and corresponding standard errors of the estimates for number of fragment impacts, total mass impact and total kinetic energy impact to the target area resulting from a 2 km intercept.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	0.067	0.067	0	Right
0	0	0.67	0.67	0	0
	0	0.29	0.29	0	
Short & Left 0		Short 2.57			Short & Right 0

Figure 6. Mean number of fragments to impact the target which is divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 2 km.

X= Downrange Distance

Y= Off-Target Distance

Z= Mean Number of Hits

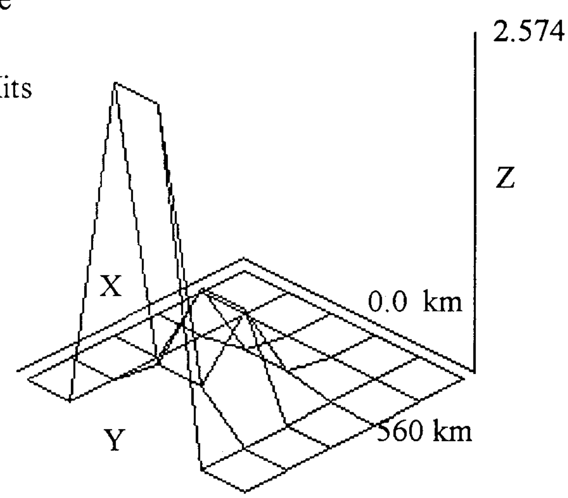


Figure 7. 3-D plot of mean number of fragment impacts per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 2 km.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	267.82	267.82	0	Right
0	0	1,258.15	1,258.15	0	0
	0	274.42	274.42	0	
Short & Left 0		Short 986.02			Short & Right 0

. ..ure 8. Mean total mass (kg) to impact the target divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 2 km.

X= Downrange Distance

Y= Off-Target Distance

Z= Mean Mass Impact

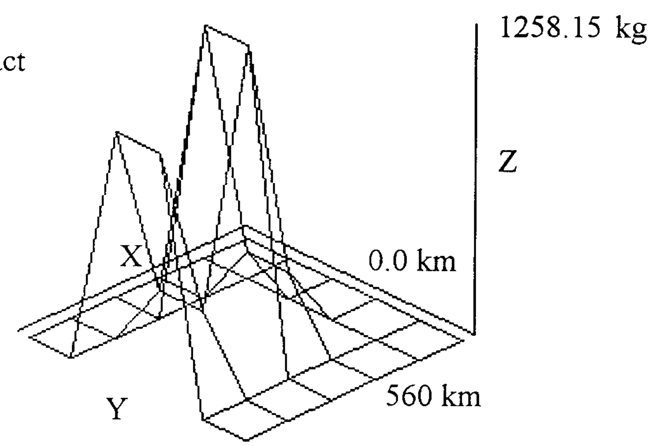


Figure 9. 3-D plot of mean total mass (kg) impact per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 2 km.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	9.036	9.036	0	Right
0	0	18.81	18.81	0	0
	0	1.624	1.624	0	
Short & Left 0		Short 546.039			Short & Right 0

Figure 10. Mean total kinetic energy (J) to impact the target divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 2 km.

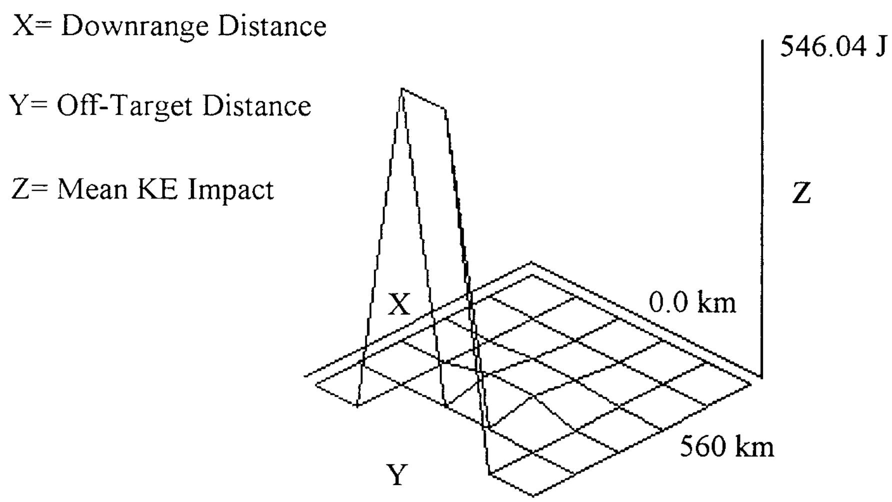


Figure 11. 3-D plot of mean kinetic energy (J) impact per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 2 km.

Block Number	Expected Value	Standard Error
10-Frag	0.273	0.007
10-Mass	822.03 kg	22.894 kg
10-K E	12.182 J	0.394 J
11-Frag	0.273	0.007
11-Mass	822.03 kg	22.894 kg
11-K E	12.182 J	0.394 J
14-Frag	0.471	0.011
14-Mass	712.543 kg	16.144 kg
14-K E	5.749 J	0.134 J
15-Frag	0.471	0.011
15-Mass	712.543 kg	16.144 kg
15-K E	5.749 J	0.134 J
Short-Frag	3.137	0.061
Short-Mass	1517.653 kg	29.438 kg
Short-K E	510.411 J	11.848 J

Table 2. List of estimates of expected values and corresponding standard errors of the estimates for number of fragment impacts, total mass impact and total kinetic energy impact to the target area resulting from a 3 km intercept.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0.273	0.273	0	0
	0	0.471	0.471	0	
Short & Left 0		Short 3.137			Short & Right 0

Figure 12. Mean number of fragments to impact the target which is divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 3 km.

X= Downrange Distance

Y= Off-Target Distance

Z= Mean Number of Hits

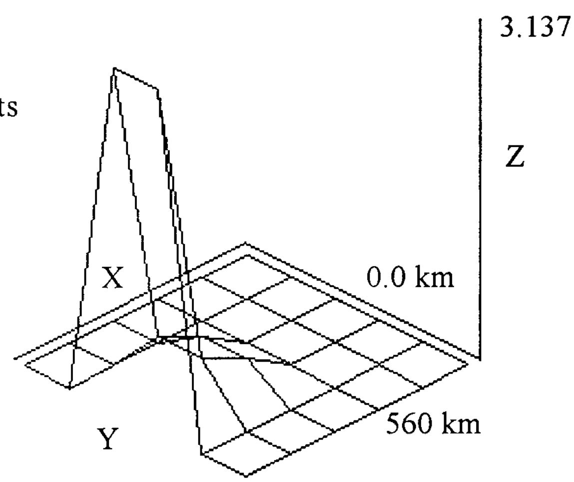


Figure 13. 3-D plot of mean number of fragment impacts per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 3 km.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	822.03	822.03	0	0
	0	712.543	712.543	0	
Short & Left 0		Short 1517.653			Short & Right 0

Figure 14. Mean total mass (kg) to impact the target divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 3 km.

X= Downrange Distance

Y= Off-Target Distance

Z= Mean Mass Impact

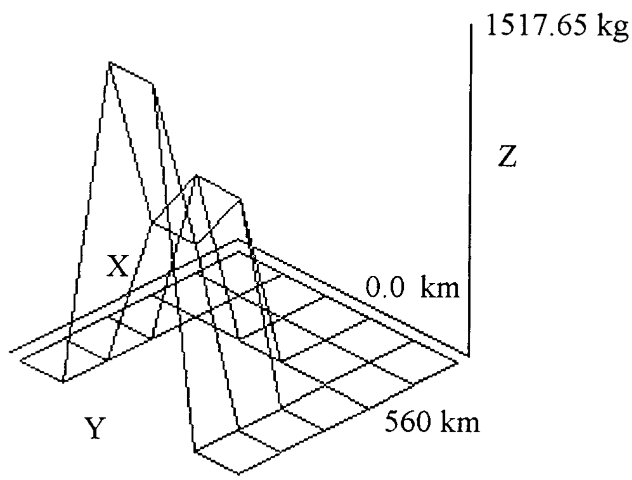


Figure 15. 3-D plot of mean total mass (kg) impact per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 3 km.

Passed & Left					Passed & Right
0		Passed 0			0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	12.182	12.182	0	0
	0	5.749	5.749	0	
Short & Left		Short			Short & Right
0		510.411			0

Figure 16. Mean total kinetic energy (J) to impact the target divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 3 km.

X= Downrange Distance

Y= Off-Target Distance

Z= Mean KE Impact

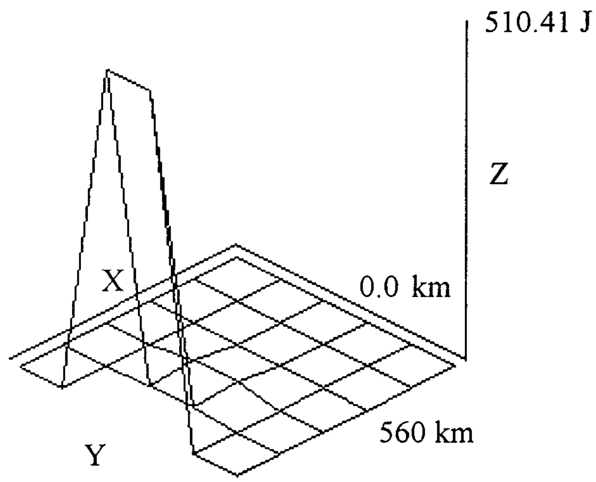


Figure 17. 3-D plot of mean kinetic energy (J) impact per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 3 km.

Block Number	Expected Value	Standard Error
10-Frag	0.017	0.002
10-Mass	88.549 kg	10.683 kg
10-K E	1.435 J	0.174 J
11-Frag	0.017	0.002
11-Mass	88.549 kg	10.683 kg
11-K E	1.435 J	0.174 J
14-Frag	0.259	0.007
14-Mass	740.46 kg	21.234 kg
14-K E	7.956 J	0.241 J
15-Frag	0.259	0.007
15-Mass	740.46 kg	21.234 kg
15-K E	7.956 J	0.241 J
Short-Frag	4.073	0.065
Short-Mass	2928.78 kg	42.802 kg
Short-K E	460.854 J	10.895 J

Table 3. List of estimates of expected values and corresponding standard errors of the estimates for number of fragment impacts, total mass impact and total kinetic energy impact to the target area resulting from a 4 km intercept.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0.017	0.017	0	0
	0	0.259	0.259	0	
Short & Left 0		Short 4.073			Short & Right 0

Figure 18. Mean number of fragments to impact the target which is divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 4 km.

X= Downrange Distance

Y= Off-Target Distance

Z= Mean Number of Hits

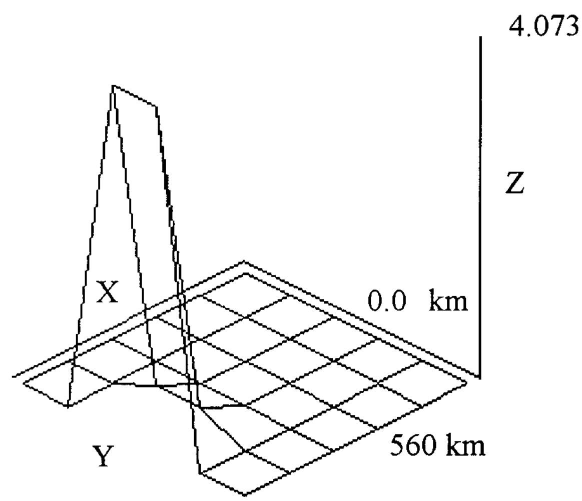


Figure 19. 3-D plot of mean number of fragment impacts per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 4 km.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	88.549	88.549	0	0
	0	740.46	740.46	0	
Short & Left 0		Short 2928.78			Short & Right 0

Figure 20. Mean total mass (kg) to impact the target divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 4 km.

X= Downrange Distance

Y= Off-Target Distance

Z= Mean Mass Impact

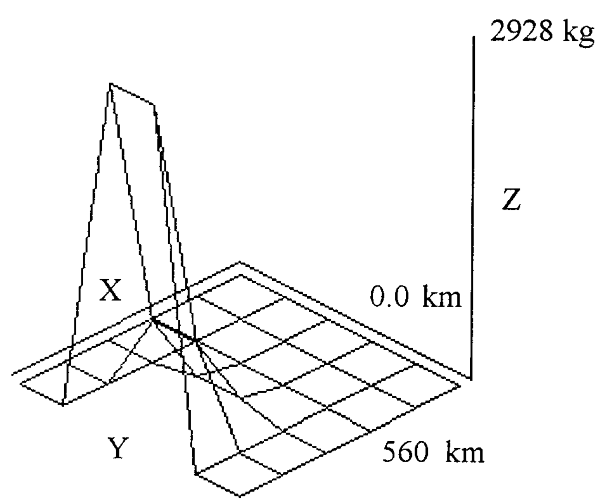


Figure 21. 3-D plot of mean total mass (kg) impact per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 4 km.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	1.435	1.435	0	0
	0	7.956	7.956	0	
Short & Left 0		Short 460.854			Short & Right 0

Figure 22. Mean total kinetic energy (J) to impact the target divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 4 km.

X= Downrange Distance

Y= Off-Target Distance

Z= Mean KE Impact

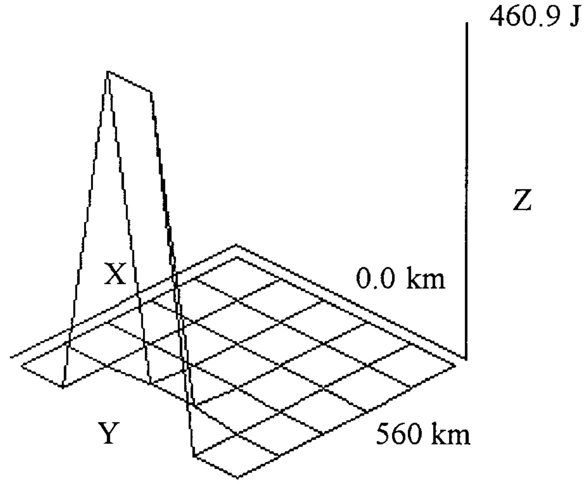


Figure 23. 3-D plot of mean kinetic energy (J) impact per target block utilizing launch angles of 30.0 degrees vertical and 0.0 degrees off-target and an intercept range of 4 km.

To aid in data interpretation, it is assumed that the target damage template encompasses the entire target region, meaning each block is a block of interest. Also, damage calculations were conducted for the surrounding areas in order to distinguish where the remaining fragments fall. An interesting result is that the total number of fragment impacts, including the surrounding areas, experienced in the three intercept ranges, as shown in Figures 4, 6, and 10, are equal. This is expected since 2000 replications are run and all impacts are tabulated. The total masses are also the same for the three intercept ranges. However, as expected, the total kinetic energies decrease as intercept range increases because the terminal velocities of the fragments are smaller. Also of note is the symmetry experienced around the target region's centerline when the off-target launch angle of 0.0° is used. This result is expected when taking a large enough sample. However, with only 2000 replications a small degree of sampling error remained. To account for this error, the damage values that resulted for the blocks on either side of the target line were averaged together and this result was applied to both blocks. This procedure is justified because of the symmetry; it would not be necessary if larger samples were taken but without doing so the computing time required would be increased unnecessarily. Any other off-target launch angle does not produce symmetry around the centerline as shown in Figure 24. The total number of fragment impacts is still the same with this new off-target launch angle, but the fragment impact positions have shifted over to the right, as anticipated. Increasing this angle more causes a further shift to the right of the fragment impact positions up to a point where no fragments impact the target region.

Determining the actual resulting damage if the TBM was left alone to impact the target region at the aim point is very complicated. Knowledge of any structures contained within the region would be necessary, along with the manner in which the blast from the warhead affects these structures. To simplify this problem, the same damage estimates used for the fragments are employed. Because four blocks surround the aim point when launch angles of 30.0° vertical and 0.0° off-target are used, the resulting damage is spread equally among the four blocks as shown in Figures 25 and 26.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	0	0.134	0	Right
0	0	0	1.34	0	0
	0	0	0.579	0	
Short & Left 0		Short 2.5735			Short & Right 0

Figure 24. Mean number of fragments to impact the target which is divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.0 degrees vertical and 0.025 degrees off-target and an intercept range of 2 km.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	1,500	1,500	0	Right
0	0	1,500	1,500	0	0
	0	0	0	0	
Short & Left 0		Short 0			Short & Right 0

Figure 25. Mean total mass to impact target region when TBM is not intercepted and aim point is impacted.

Passed & Left 0		Passed 0			Passed & Right 0
	0	0	0	0	
Left	0	1,407.216	1,407.216	0	Right
0	0	1,407.216	1,407.216	0	0
	0	0	0	0	
Short & Left 0		Short 0			Short & Right 0

Figure 26. Mean total kinetic energy to impact target region when TBM is not intercepted and aim point is impacted

V. DISCUSSION AND CONCLUSIONS

The estimates of damage occurring to the target region vary depending on three factors. The first is the vertical launch angle. As this angle is increased the number of fragments that impact the target region increases, along with the respective masses, since the TBM is at a higher altitude at intercept than when a smaller angle is employed. This is shown in Figures 6 and 27. The higher altitude produces a narrower dispersion pattern which concentrates the fragments impacts closer to the target. However, the kinetic

Passed & Left 0		Passed 0			Passed & Right 0
	0	0.263	0.263	0	
Left	0	0.571	0.571	0	Right
0	0	0.286	0.286	0	0
	0	0.161	0.161	0	
Short & Left 0		Short 2.066			Short & Right 0

Figure 27. Mean number of fragments to impact the target which is divided into 16 equal blocks and its surrounding areas. This data is generated using launch angles of 30.1 degrees vertical and 0.0 degrees off-target and an intercept range of 2 km.

energies are smaller because the fragments fly longer than when intercepted at a lower altitude thus allowing air resistance more time to reduce the velocities. The second factor is the off-target launch angle. As this angle is increased, i.e., moved to the right of target line, the extent of damage to the target region shifts to the right up to a point where all the fragments miss the target region. A shift to the left occurs when this angle is decreased. i.e., moved to the left of the target line. Since the off-target component of velocity is treated equally in both cases, moving right or left of the target line, the extent of damage is equal, however, the block which suffers the damage is transposed to the opposite side of the target line. As shown in Appendix B, 0.15° is the angle which shifts everything outside of the target region. The shift is shown in Figures 6 and 24. The last factor which causes variance in the damage estimates is the intercept range. Intercepts which occur close to the aim point generate more fragment impacts to the target region than intercepts which are not as close. The difference is illustrated in Figures 6 and 18.

These three factors, taken independently, yield fairly distinct distributions of damage. However, when examined simultaneously, these factors create a very diverse damage distribution. Appendix B contains the layouts of all possible combinations of these three factors for the example given in the last chapter. The only exceptions are the off-target angles to the left of the target line. The extent of damage is the same, but the blocks in which the damage occurs are transposed on the opposite side of the target line. One possible approach to estimating the extent of damage when these three factors are known is to specify a *response function*,

$$\textit{Expected Damage} = f(\Theta_{OT}, \Theta_V, R) \tag{26}$$

where

Θ_{OT} = off-target launch angle
Θ_V = vertical launch angle
R = intercept range measured in km downrange from the target

Such a function, specified on physical grounds, could be fitted to data obtained at various angles and ranges, and then used to estimate responses at others.

The conclusion is that knowing the expected damage to occur to a target region if a TBM is intercepted is helpful in making a decision whether to intercept it or not. This is obvious, but the model is developed to illustrate the necessity of this information. This model is a simple analytical attempt to produce a tentative estimate. At this time, very little is known concerning the dispersion of fragments generated by an intercept. Because of this lack of information, a crude estimate is the only estimate available. When the assumptions made in this thesis are determined with certainty, a more detailed version of the model, replicating the exact flight paths and impact points, can be developed.

Another area that may be of interest is determining which type of intercept approach is better, a blast-fragment intercept or a kinetic energy intercept. A blast-fragment intercept such as the one simulated in this thesis does not require the accuracy of a kinetic energy intercept where the interceptor must actually impact the TBM. However, the kinetic energy intercept almost completely destroys the TBM in flight resulting in very little chance of collateral damage. The technology required for a kinetic energy interceptor is more costly than that of the blast-fragment interceptor, and because of this high cost it does not presently exist. The Navy is experimenting with the Light Exo-Atmospheric Projectile (LEAP), a kinetic kill vehicle that impacts a TBM outside of the atmosphere. If this experiment is successful the technology may be applied to work within the atmosphere on intercepts such as the ones proposed in this thesis eliminating the need for an estimate of fragment damage.

APPENDIX A.

CALCULATION OF TBM FLIGHT PATH

As mentioned in the text, only four of the actual characteristics of the Al Hussein are used in this model: the TBM's mass, the TBM's length, the TBM's cross-sectional area, and the TBM's boost-phase time interval. The mass of the TBM at launch is 7000 kg. However, at the end of boost phase the mass is reduced to 6000 kg due to expended fuel. The method for compensating for expended fuel is not entirely realistic since the mass does not just disappear all at once. However, since the boost phase is not of concern in this thesis, and since the assumption has been made that straight-line motion during boost was unaffected by air resistance and gravitation, the manner in which missile mass is decreased during fuel use is insignificant so long as the mass at intercept equals the mass at the completion of boost phase. The length of the TBM is 12 meters. The diameter of the TBM is 0.88 meters at its widest point yielding a cross-sectional area of 0.6082 meters^2. Finally, determining the exact boost-phase time interval is extremely difficult and requires numerous factors, especially the fuel burn rate of the TBM which depends heavily upon the density and temperature of air. Usually, as is the case with the Al Hussein, the length of boost phase time interval is given by an approximation. Because of this complexity in determining the exact time interval, the approximate boost phase time interval of 90 seconds is used in this model [Ref. 2].

To determine the remaining parameters, an assumption about the distance to the target region is required; it is that the center of the target region designated for attack, i.e., the aim point, is located 560 km from the launch point. Having knowledge of the target distance and choosing an arbitrary launch angle of 30 degrees enables the launch acceleration to be calculated with the assumption that acceleration was constant throughout the entire boost phase. The calculated launch acceleration is 0.023875 meters/seconds^2. The known parameters, together with these parameters calculated

APPENDIX A.

CALCULATION OF TBM FLIGHT PATH

under the above assumptions, are the only data required to model the launch of the TBM and begin the simulation. Since these calculated characteristics are not the precise characteristics of the Al Hussein TBM, the flight paths modeled in this simulation do not replicate those of an Al Hussein exactly, and thus, are considered generic TBM flight paths usefully close to the actual to provide insight. All of the above numbers can be changed in the software at the will of an analyst.

TBM Flight Path Calculations

Prior to commencing the simulation, the user is prompted for three items that aid in the construction of the simulated TBM flight path. The first item is the range at which intercept is going to occur, measured in kilometers downrange from the aim point. The last two items, the vertical and the off-target launch angles, determine the location of the TBM's impact point if it is not intercepted. In this thesis only one pair of angles, 30 degrees vertical and 0 degrees off-target, result in a direct hit of the aim point. All other pairs yield different impact points that are off-target-region center. Launching a TBM, even one with state-of-the-art technology, at a target 560 km away results in some level of uncertainty as to the exact impact point. Of course, better technology leads to smaller uncertainty. Varying the vertical and off-target launch angles slightly, i.e., up to +/- 0.3 degrees, allows parametric sensitivity analysis to be conducted in this model. Such numbers are illustrative and can be altered by the analyst.

The TBM flight is broken up into two segments. The first segment, which consists of the boost phase, is fairly easy to simulate taking into account the assumption of straight-line motion, during which air resistance and gravity have negligible effect. Basic equations of physics are employed to determine the TBM's position and velocity at the

APPENDIX A.

CALCULATION OF TBM FLIGHT PATH

completion of boost phase. The linear distance traveled by the TBM during boost phase, D, is determined from a side calculation where gravity and thrust are taken into account. From this the three-dimensional position of the TBM at the completion of the boost phase is calculated using the following equations:

$$X = D \cos(LA) \sin(OA) \tag{1}$$

$$Y = D \sin(LA) \tag{2}$$

$$Z = D \cos(LA) \cos(OA) \tag{3}$$

where

X = downrange displacement from launch point

Y = altitude

Z = off-target displacement

LA = vertical launch angle

OA = off-target angle

The TBM's velocity in the direction tangent to flight path after boost phase, V_0, is also determined from a side calculation where gravity and thrust are taken into account. From this the components of velocity are calculated in the following manner

$$V_X = V_0 \cos(LA) \cos(OA) \tag{4}$$

$$V_Y = V_0 \sin(LA) \tag{5}$$

APPENDIX A.

CALCULATION OF TBM FLIGHT PATH

$$V_Z = V_0 \cos(LA) \sin(OA) \tag{6}$$

where

V_X = downrange component of velocity

V_Y = vertical component of velocity

V_Z = off-target component of velocity

The second segment consists of the TBM ballistic flight path. The purpose for its generation is to enable examination of various intercept points so as to determine the lethality of the fragments created by those intercepts to parts of the target region. Therefore, it is necessary to be able to calculate the positions and velocities of the TBM throughout its entire flight path. Initially, an attempt is made to replicate the exact ballistic path the TBM would follow through the atmosphere. Knowing that gravity and air resistance are the only two forces which affect a projectile's flight, an expression of air resistance is sought to implement in a closed-form differential equation of motion. It has been determined that retardation of a projectile due to air resistance is a function of the projectile's velocity [Ref. 3].

The actual flight of the TBM is modeled using three-dimensional equations of motion taking gravity and atmospheric density into consideration. In the simulation, a loop is created which continues until the downrange displacement of the TBM equals the intercept range entered by the user. The position and velocity of the TBM, calculated by Equations (1) through (6), are the initial conditions prior to entering the loop. For ease of computation, an interval of one second is chosen as the time step in the loop. At each

APPENDIX A.

CALCULATION OF TBM FLIGHT PATH

time step the TBM's terminal velocity at time t is calculated by

$$V_T(t) = \sqrt{\frac{2mg}{C\rho(t)A}} \tag{7}$$

where

- m = mass of the TBM
- g = acceleration due to gravity
- C = drag coefficient dependent on angle of attack
- $\rho(t)$ = density of air which is a function of altitude at time t
- A = cross-sectional area of the TBM

Since the remaining equations are all functions of time, the reference to time, i.e., (t), is a given and thus removed from the equations. The velocity of the TBM at each pass through the loop is updated using

$$V = (V_X^2 + V_Y^2 + V_Z^2)^{\frac{1}{2}} \tag{8}$$

From these two calculations, the changes in each component of velocity are calculated using the following equations of motion

$$\frac{dV_X}{dt} = -\frac{VV_X g}{V_T^2} \tag{9}$$

APPENDIX A.

CALCULATION OF TBM FLIGHT PATH

$$\frac{dV_Y}{dt} = -g - \frac{V V_Y g}{V_T^2} \tag{10}$$

$$\frac{dV_Z}{dt} = -\frac{V V_Z g}{V_T^2} \tag{11}$$

From these equations much simpler forms for finding the velocity components at each (1-second) time step are defined by

$$V_{X_{t+\Delta}} = V_{X_t} - \frac{V V_{X_t} C \rho A}{2m} \tag{12}$$

$$V_{Y_{t+\Delta}} = V_{Y_t} - g - \frac{V V_{Y_t} C \rho A}{2m} \tag{13}$$

$$V_{Z_{t+\Delta}} = V_{Z_t} - \frac{V V_{Z_t} C \rho A}{2m} \tag{14}$$

The simple one-step differential equation solver portrayed above can be replaced by far more accurate numerical procedures, such as 4th-order Runge-Kutta, or other methodologies in standard package programs [Ref. 4]. In view of the exploratory nature of this investigation such complexity has not been introduced.

Atmospheric density, ρ, is extremely difficult to model. The Earth's atmosphere is primarily composed of nitrogen and oxygen. Solar radiation affects the dynamic properties of this medium by constantly changing the temperature, pressure, and chemical constituents, particulate presence and electrical properties [Ref. 5]. The inability to model

APPENDIX A.

CALCULATION OF TBM FLIGHT PATH

atmospheric density creates a problem when trying to replicate the exact flight path of the TBM. Therefore, the values for ρ utilized in this model are taken from the U.S. Standard Atmosphere Table [Ref. 6]. The drag coefficient, C, takes into account the relative contributions of viscous and form resistances, and it depends on the nature of the TBM (size, shape, and irregularity and roughness of its surface) as well as on the characteristics of the flow of air over the TBM [Ref. 7]. Numerous factors are involved in computing C; a value of 1.0 is used in the present illustrative simulation.

The position of the TBM is also required at every step in the loop. Having calculated the velocity, the position can be determined rather easily by

$$X_t = X_{t-\Delta} + V_{X_t} \tag{15}$$

$$Y_t = Y_{t-\Delta} + V_{Y_t} \tag{16}$$

$$Z_t = Z_{t-\Delta} + V_{Z_t} \tag{17}$$

The loop continues until the downrange displacement, X_t, equals the intercept point determined by the user. When the intercept point is reached the TBM separates into a random number of fragments.

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 4.626 4589.799 1268.729			S & R 0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.0°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

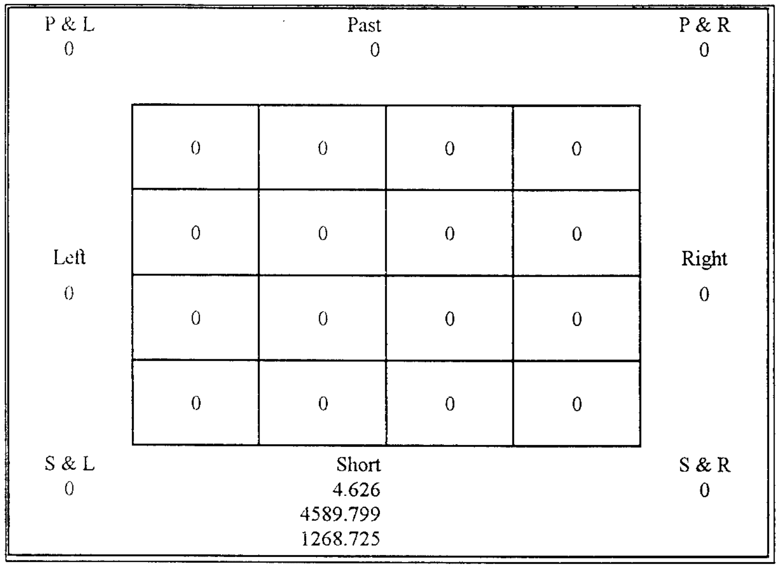
P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 4.626 4589.799 1268.729			S & R 0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.025°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

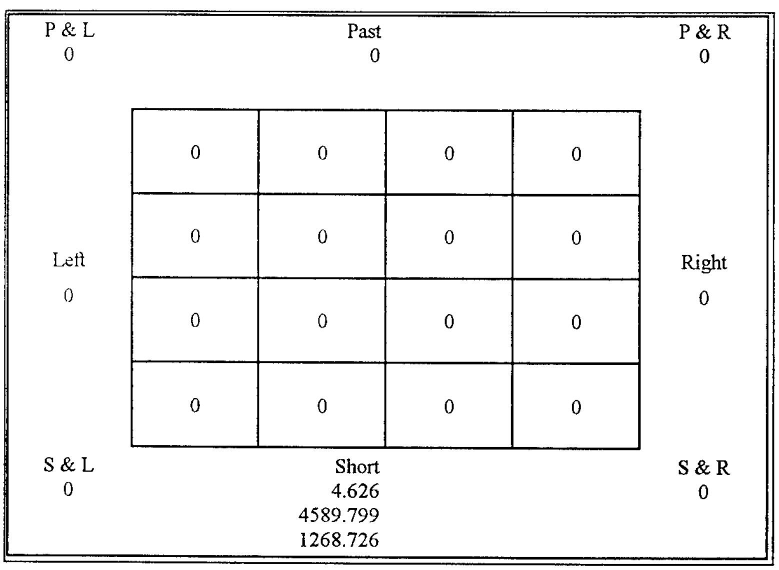
APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES



Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.05°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE



Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.075°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L
0

Past
0

P & R
0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Left
0

Right
0

S & L
0

Short
4.626
4589.799
1268.726

S & R
0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.10°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L
0

Past
0

P & R
0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Left
0

Right
0

S & L
0

Short
0

S & R
4.626
4589.799
1268.726

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.15°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0.098 300.463 19.300	0.098 300.463 19.300	0	
	0	0.996 1551.613 65.699	0.996 1551.613 65.699	0	
S & L 0		Short 2.438 882.646 445.168			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.0°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0.195 600.927 38.599	0	
	0	0	1.992 3102.865 131.398	0	
S & L 0		Short 2.439 883.007 445.170			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.025°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0.194 598.565 38.490	0	0
	0	0	1.993 3105.227 131.507	0	
S & L 0		Short 2.439 883.007 445.170			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.05°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0.193 596.314 38.319	0
	0	0	0	1.994 3107.478 131.676	
S & L 0		Short 2.439 883.007 445.170			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.075°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L		Past		P & R
0		0		0

0	0	0	0
0	0	0	0
0	0	0	0.192 592.009 38.152
0	0	0	1.995 3111.418 131.841

Left: 0

Right: 0

S & L		Short		S & R
0		2.439 883.371 445.171		0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.10°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L		Past		P & R
0		0		0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Left: 0

Right: 2.185
3701.956
169.988

S & L		Short		S & R
0		0		2.441 884.843 445.177

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.15°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0
0	0.0668 267.819 9.036	0.0668 267.819 9.036	0
0	0.670 1258.149 18.810	0.670 1258.149 18.810	0
0	0.290 274.421 1.624	0.290 274.421 1.624	0

S & L 0 | Short 2.574 986.021 546.039 | S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.0°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0
0	0	0.134 535.637 18.071	0
0	0	1.340 2516.299 37.621	0
0	0	0.579 548.842 3.248	0

S & L 0 | Short 2.574 986.021 546.039 | S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.025°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0.115 459.940 15.399	0.018 74.201 2.640	Right 0
	0	0	1.264 2341.621 34.498	0.076 176.174 3.154	
	0	0	0.579 548.842 3.248	0	
S & L 0		Short 2.574 986.021 546.039			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.05°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0.133 531.701 17.930	Right 0
	0	0	0	1.340 2519.113 37.755	
	0	0	0	0.580 549.963 3.255	
S & L 0		Short 2.574 986.021 546.039			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.075°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0.133 531.701 17.930	Right 0
	0	0	0	1.340 2519.113 37.755	
	0	0	0	0.580 549.963 3.255	
S & L 0		Short 2.574 986.021 546.039			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.10°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 2.052 3600.371 58.938
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 0			S & R 2.574 986.428 58.938

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.15°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0.263 798.245 14.709	0.263 798.245 14.709	0
0	0.571 830.038 7.203	0.571 830.038 7.203	0
0	0.286 242.519 1.430	0.286 242.519 1.430	0
0	0.161 104.630 0.555	0.161 104.630 0.555	0

S & L 0 | Short 2.066 635.936 437.729 | S & R 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.0°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0.526 1596.489 29.418	0
0	0	1.141 1660.075 14.406	0
0	0	0.573 485.04 2.860	0
0	0	0.321 209.261 1.110	0

S & L 0 | Short 2.066 635.936 437.729 | S & R 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.025°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L: 0 | Past: 0 | P & R: 0

Left: 0 | Right: 0

0	0	0.419 1259.964 22.890	0.107 336.525 6.528
0	0	1.01 1456.676 12.555	0.13 202.409 1.845
0	0	0.571 483.062 2.847	0.003 2.966 0.019
0	0	0.320 208.661 1.107	0

S & L: 0 | Short: 2.067
636.536
437.732 | S & R: 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.05°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L: 0 | Past: 0 | P & R: 0

Left: 0 | Right: 0

0	0	0	0.525 1595.345 29.403
0	0	0	1.141 1660.228 14.414
0	0	0	0.574 486.028 2.866
0	0	0	0.320 208.366 1.106

S & L: 0 | Short: 2.067
636.840
437.734 | S & R: 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.075°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0.525 1595.358 29.403
0	0	0	1.139 1658.697 14.404
0	0	0	0.575 487.189 2.874
0	0	0	0.320 208.415 1.106

S & L 0 | Short 2.068 637.140 437.735 | S & R 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.10°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

Left 0 | Right 2.558 3949.358 47.785

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

S & L 0 | Short 0 | S & R 2.068 637.441 437.737

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.15°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0	Past 0.012 67.252 1.084	P & R 0
Left 0	(grid below)	Right 0
S & L 0	Short 2.520 943.921 600.996	S & R 0

0	0.194 624.831 7.073	0.194 624.831 7.073	0
0	0.350 634.771 5.117	0.350 634.771 5.117	0
0	0.306 357.063 2.402	0.306 357.063 2.402	0
0	0.198 171.147 1.041	0.198 171.147 1.041	0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.0°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0	Past 0..012 67.252 1.084	P & R 0
Left 0	(grid below)	Right 0
S & L 0	Short 2.520 943.921 600.996	S & R 0

0	0	0.388 1249.663 14.145	0
0	0	0.700 1268.838 10.230	0
0	0	0.612 714.831 4.809	0
0	0	0.395 342.293 2.081	0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.025°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0.012 67.252 1.084			P & R 0
	0	0	0.280 899.583 10.148	0.108 350.080 3.997	
Left 0	0	0	0.552 997.606 8.030	0.147 270.526 2.195	Right 0
	0	0	0.537 624.784 4.198	0.076 90.753 0.616	
	0	0	0.391 338.543 2.058	0.004 3.750 0.023	
S & L 0		Short 2.520 943.921 600.996			S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.05°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0.012 64.276 1.030			P & R 0
	0	0	0	0.388 1252.639 14.199	
Left 0	0	0	0	0.699 1268.132 10.224	Right 0
	0	0	0	0.612 715.046 4.811	
	0	0	0	0.395 342.396 2.082	
S & L 0		Short 2.520 944.309 600.999			S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.075°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0.010 53.552 0.864			P & R 0.002 8.073 0.125
	0	0	0	0.366 1181.029 13.379	
Left	0	0	0	0.699 1268.237 10.226	Right
0	0	0	0	0.612 714.565 4.807	0.023 74.157 0.860
	0	0	0	0.395 342.498 2.083	
S & L 0		Short 2.521 944.688 601.000			S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.10°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0.010 56.533 0.914
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0	2.094 3584.432 31.416
	0	0	0	0	
S & L 0		Short 0			S & R 2.522 945.833 600.719

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.15°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0.812 2144.979 22.719			P & R 0
	0	0.380 506.979 3.607	0.380 506.979 3.607	0	
Left 0	0	0.267 241.988 1.491	0.267 241.988 1.491	0	Right 0
	0	0.176 122.419 0.692	0.176 122.419 0.692	0	
	0	0.117 69.476 0.374	0.117 69.476 0.374	0	
S & L 0		Short 1.935 560.094 463.798			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.0°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0.812 2144.979 22.719			P & R 0
	0	0	0.760 1013.958 7.215	0	
Left 0	0	0	0.535 483.977 2.983	0	Right 0
	0	0	0.352 244.839 1.385	0	
	0	0	0.234 138.952 0.749	0	
S & L 0		Short 1.935 560.094 463.798			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.025°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0.811 2144.134 22.713			P & R 0
	0	0	0.607 805.824 5.722	0.153 208.449 1.496	
Left	0	0	0.492 443.053 2.726	0.043 40.674 0.255	Right
0	0	0	0.353 245.617 1.389	0	0
	0	0	0.234 138.952 0.749	0	
S & L 0		Short 1.935 560.094 463.798			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.05°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0.811 2144.134 22.7130			P & R 0
	0	0	0	0.759 1013.205 7.211	
Left	0	0	0	0.535 484.796 2.988	Right
0	0	0	0	0.353 245.617 1.389	0
	0	0	0	0.233 138.672 0.747	
S & L 0		Short 1.935 560.375 463.799			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.075°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0.765 1998.791 20.992			P & R 0.046 145.342 1.720
	0	0	0	0.758 1012.162 7.204	
Left	0	0	0	0.535 485.449 2.993	Right
0	0	0	0	0.353 246.007 1.391	0
	0	0	0	0.233 138.392 0.746	
S & L 0		Short 1.936 560.655 463.801			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.10°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0.810 2141.600 22.693
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0	1.880 1884.266 12.353
	0	0	0	0	
S & L 0		Short 0			S & R 1.936 560.932 463.802

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.15°
Intercept Range = 2.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L		Past		P & R
0		0		0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Left 0 — Right 0

S & L	Short	S & R
0	4.626 4589.799 563.059	0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.0°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L		Past		P & R
0		0		0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Left 0 — Right 0

S & L	Short	S & R
0	4.626 4586.799 563.609	0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.025°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 4.626 4586.799 563.609			S & R 0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.05°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 4.626 4586.799 563.678			S & R 0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.075°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 4.602 4526.661 561.469			S & R 0.024 60.138 2.209

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.10°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 0			S & R 4.626 4586.799 563.714

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.15°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0.446 1136.970 22.202	0.446 1136.970 22.202	0	
S & L 0		Short 3.735 2312.858 685.748			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.0°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0.892 2274.313 44.314	0	
S & L 0		Short 3.734 2312.485 684.104			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.025°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0.721 1826.456 35.391	0.171 447.857 8.923	
S & L 0		Short 3.734 2312.485 684.104			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.05°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0.891 2273.502 44.306	
S & L 0		Short 3.735 2313.296 684.112			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.075°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	
	0	0	0	0	Right 0.042 117.494 2.560
	0	0	0	0.849 2154.406 41.732	
S & L 0		Short 3.729 2307.361 684.010			S & R 0.007 7.537 0.117

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.10°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	
	0	0	0	0	Right 0.887 2266.316 44.238
	0	0	0	0	
S & L 0		Short 0			S & R 3.739 2320.483 684.178

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.15°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0.273 822.030 12.182	0.273 822.030 12.182	0	0
	0	0.471 712.543 5.749	0.471 712.543 5.749	0	
S & L 0		Short 3.137 1517.653 510.411			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.0°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0.541 1631.606 24.248	0	0
	0	0	0.943 1431.659 11.595	0	
S & L 0		Short 3.142 1523.534 508.461			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.025°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0.392 1177.263 17.471	0.149 454.342 6.777	0
	0	0	0.739 1116.472 9.007	0.204 315.187 2.589	
S & L 0		Short 3.142 1523.534 508.461			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.05°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0.540 1629.495 24.227	0
	0	0	0	0.943 1433.208 11.613	
S & L 0		Short 3.143 1524.095 508.465			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.075°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0.473 1424.101 21.152	0.129 301.653 3.870
	0	0	0	0.880 1335.286 10.807	
S & L 0		Short 3.133 1515.166 508.376			S & R 0.011 10.593 0.100

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.10°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0	1.483 3062.785 35.826
	0	0	0	0	
S & L 0		Short 0			S & R 3.143 1524.014 508.466

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.15°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L	Past	P & R
0	0	0

Left: 0 | Right: 0

0	0.002 12.496 0.224	0.002 12.496 0.224	0
0	0.212 679.052 8.387	0.212 679.052 8.387	0
0	0.355 627.812 5.082	0.355 627.812 5.082	0
0	0.284 327.774 2.182	0.284 327.774 2.182	0

S & L	Short	S & R
0	2.921 1292.530 679.329	0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.0°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L	Past	P & R
0	0	0

Left: 0 | Right: 0

0	0	0.006 30.456 0.544	0
0	0	0.425 1358.475 16.731	0
0	0	0.714 1258.245 10.174	0
0	0	0.564 649.800 4.322	0

S & L	Short	S & R
0	2.918 1289.823 680.669	0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.025°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0	Past 0	P & R 0
Left 0	(grid below)	Right 0
S & L 0	Short 2.918 1289.823 680.669	S & R 0

0	0	0.005 24.895 0.444	0.001 5.561 0.010
0	0	0.296 943.509 11.621	0.129 414.966 5.110
0	0	0.540 944.799 7.608	0.174 313.446 2.566
0	0	0.460 529.408 3.520	0.104 120.392 0.803

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.05°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0	Past 0	P & R 0
Left 0	(grid below)	Right 0
S & L 0	Short 2.918 1289.823 680.669	S & R 0

0	0	0	0.006 30.456 0.544
0	0	0	0.424 1356.261 16.711
0	0	0	0.715 1260.459 10.194
0	0	0	0.564 649.800 4.322

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.075°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0.005 27.639 0.493	
Left 0	0	0	0	0.360 1147.004 14.133	Right 0.158 365.585 3.848
	0	0	0	0.645 1134.862 9.163	
	0	0	0	0.540 621.880 4.135	
S & L 0		Short 2.917 1289.748 680.602			S & R 0.001 0.081 0.068

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.10°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 1.707 3295.989 31.766
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 0			S & R 2.919 1290.809 680.676

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.15°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0.014 75.944 1.007	0.014 75.944 1.007	0
0	0.138 463.670 4.787	0.138 463.670 4.787	0
0	0.284 583.692 4.834	0.284 583.692 4.834	0
0	0.332 437.512 3.085	0.332 437.512 3.085	0

S & L 0 | Short 3.089 1465.159 532.897 | S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.0°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0.030 158.466 2.084	0
0	0	0.273 915.140 9.444	0
0	0	0.563 1159.855 9.622	0
0	0	0.675 891.681 6.291	0

S & L 0 | Short 3.086 1461.656 534.408 | S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.025°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0.021 111.412 1.469	0.009 47.054 0.615	
Left 0	0	0	0.177 591.874 6.103	0.096 323.266 3.341	Right 0
	0	0	0.396 812.463 6.729	0.167 347.391 2.893	
	0	0	0.506 667.211 4.705	0.169 224.470 1.586	
S & L 0		Short 23.086 1461.656 534.408			S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.05°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0.030 158.466 2.084	
Left 0	0	0	0	0.272 912.522 9.421	Right 0
	0	0	0	0.564 1162.473 9.646	
	0	0	0	0.675 891.681 6.291	
S & L 0		Short 3.086 1461.656 534.408			S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.075°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0.025 132.006 1.735	
	0	0	0	0.213 714.238 7.366	
Left 0	0	0	0	0.474 975.054 8.085	Right 0.23 514.430 4.688
	0	0	0	0.598 788.876 5.563	
S & L 0		Short 3.054 1431.829 534.221			S & R 0.033 30.365 0.191

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.10°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
	0	0	0	0	
Left 0	0	0	0	0	Right 1.538 3123.044 27.428
	0	0	0	0	
S & L 0		Short 0			S & R 3.088 1463.755 534.422

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.15°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0

Past 0.138 585.269 6.997

P & R 0

0	0.261 643.481 5.806	0.261 643.481 5.806	0
0	0.349 523.538 3.865	0.349 523.538 3.865	0
0	0.317 316.087 2.028	0.317 316.087 2.028	0
0	0.228 164.516 0.949	0.228 164.516 0.949	0

Left 0

Right 0

S & L 0

Short 2.180 706.286 411.460

S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.0°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0

Past 0.145 608.247 7.234

P & R 0

0	0	0.519 1271.030 11.435	0
0	0	0.697 1042.944 7.696	0
0	0	0.634 631.608 4.050	0
0	0	0.453 326.832 1.885	0

Left 0

Right 0

S & L 0

Short 2.179 706.137 410.363

S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.025°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0.145 606.546 7.216			P & R 0
	0	0	0.342 835.771 7.514	0.178 436.960 3.939	
Left 0	0	0	0.501 748.146 5.516	0.196 294.797 2.180	Right 0
	0	0	0.469 466.705 2.992	0.165 164.903 1.058	
	0	0	0.385 276.951 0.1.596	0.068 49.881 0.289	
S & L 0		Short 2.179 706.137 410.363			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.05°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0.145 606.546 7.216			P & R 0
	0	0	0	0.519 1271.755 11.445	
Left 0	0	0	0	0.697 1042.724 7.696	Right 0
	0	0	0	0.635 632.804 4.058	
	0	0	0	0.451 325.886 0.1.880	
S & L 0		Short 2.181 707.083 410.368			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.075°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0.109 459.347 5.484			P & R 0.036 147.199 1.732
	0	0	0	0.417 1019.672 9.162	
Left 0	0	0	0	0.595 887.557 6.544	Right 0.277 480.236 3.903
	0	0	0	0.571 566.795 3.630	
	0	0	0	0.442 318.910 1.839	
S & L 0		Short 2.178 705.236 410.358			S & R 0.003 1.848 0.010

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.10°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0.144 603.139 7.181
	0	0	0	0	
Left 0	0	0	0	0	Right 2.300 3275.944 25.11
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 0			S & R 2.182 707.715 410.372

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.15°
Intercept Range = 3.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

S & L 0 | Short 4.626 4586.799 473.601 | S & R 0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.0°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

S & L 0 | Short 4.626 4586.799 473.601 | S & R 0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.025°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 4.626 4586.799 473.601			S & R 0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.05°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 4.626 4586.799 473.601			S & R 0

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.075°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L		Past		P & R
0		0		0

	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0	0
	0	0	0	0	

S & L	Short	S & R
0	4.535	0.091
	4377.117	209.681
	470.248	3.352

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.10°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L		Past		P & R
0		0		0

	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0	0
	0	0	0	0	

S & L	Short	S & R
0	0	4.626
		4586.799
		473.602

Vertical Launch Angle = 29.8°
Off-Target Launch Angle = 0.15°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0.070 295.114 5.361	0.070 295.114 5.361	0	
S & L 0		Short 4.485 3996.570 679.218			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.0°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0
	0	0	0	0	
	0	0	0.141 591.754 10.744	0	
S & L 0		Short 4.485 3995.045 679.197			S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.025°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0.010 419.487 7.675	0.042 172.266 3.069

S & L 0 | Short 4.485 3995.045 679.197 | S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.05°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0.140 588.109 10.686

S & L 0 | Short 4.486 3998.690 679.255 | S & R 0

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.075°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0.017 72.074 1.316
	0	0	0	0	
	0	0	0	0.123 516.035 9.370	
S & L 0		Short 4.401 3821.709 677.376			S & R 0.085 176.981 1.879

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.10°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0	0	Right 0.139 585.089 10.650
	0	0	0	0	
	0	0	0	0	
S & L 0		Short 0			S & R 4.487 4001.710 679.290

Vertical Launch Angle = 29.9°
Off-Target Launch Angle = 0.15°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0.017 88.549 1.435	0.017 88.549 1.435	0	0
	0	0.259 740.460 7.956	0.259 740.460 7.956	0	
S & L 0		Short 4.073 2928.780 460.854			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.0°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0.032 164.373 2.643	0	0
	0	0	0.525 1501.472 16.221	0	
S & L 0		Short 4.069 2920.954 466.209			S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.025°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0
0	0	0	0
0	0	0.023 118.939 1.921	0.009 45.434 0.722
0	0	0.364 1039.273 11.208	0.165 462.199 5.013

S & L 0 | Short 4.069 2920.954 466.509 | S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.05°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0	0
0	0	0	0
0	0	0	0.032 164.373 2.643
0	0	0	0.524 1500.410 16.212

S & L 0 | Short 4.070 2922.016 466.218 | S & R 0

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.075°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0.029 145.949 2.340	0.083 253.404 2.892
	0	0	0	0.445 1265.430 13.623	
S & L 0		Short 3.946 2745.598 464.870			S & R 0.124 176.418 1.348

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.10°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0	Right
0	0	0	0	0	0.554 1660.614 18.819
	0	0	0	0	
S & L 0		Short 0			S & R 4.072 2926.185 466.254

Vertical Launch Angle = 30.0°
Off-Target Launch Angle = 0.15°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0.020 100.606 1.385	0.020 100.606 1.385	0	Right 0
	0	0.177 551.304 5.686	0.177 551.304 5.686	0	
	0	0.312 582.163 4.678	0.312 582.163 4.678	0	
S & L 0		Short 2.609 2118.654 634.174			S & R 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.0°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left 0	0	0	0.040 201.212 2.770	0	Right 0
	0	0	0.355 1102.608 11.372	0	
	0	0	0.623 1164.325 9.355	0	
S & L 0		Short 3.609 2118.654 634.174			S & R 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.025°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0.028 140.454 1.936	0.012 60.757 0.834	Right
0	0	0	0.241 745.640 7.670	0.114 356.967 3.703	0
	0	0	0.456 843.933 6.756	0.168 320.392 2.600	
S & L 0		Short 3.609 2118.654 634.174			S & R 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.05°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
Left	0	0	0	0.040 201.212 2.770	Right
0	0	0	0	0.354 1101.383 11.362	0
	0	0	0	0.624 1165.550 9.366	
S & L 0		Short 3.609 2118.654 634.174			S & R 0

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.075°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
	0	0	0	0.034 173.471 2.389	
Left 0	0	0	0	0.286 885.066 9.112	Right 0.155 397.832 3.877
	0	0	0	0.543 1011.775 8.119	
S & L 0		Short 3.547 2044.002 633.624			S & R 0.062 74.652 0.551

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.10°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
	0	0	0	0	
Left 0	0	0	0	0	Right 1.014 2463.733 23.465
	0	0	0	0	
S & L 0		Short 0			S & R 3.612 2123.036 634.207

Vertical Launch Angle = 30.1°
Off-Target Launch Angle = 0.15°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

0	0	0	0
0	0.012 66.104 0.812	0.012 66.104 0.812	0
0	0.098 355.413 3.655	0.098 355.413 3.655	0
0	0.228 529.662 4.567	0.228 529.662 4.567	0

Left 0 | Right 0

S & L 0 | Short 3.950 2684.442 471.931 | S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.0°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

0	0	0	0
0	0	0.025 132.207 1.624	0
0	0	0.196 710.826 7.310	0
0	0	0.455 1058.373 9.126	0

Left 0 | Right 0

S & L 0 | Short 3.951 2685.392 471.939 | S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.025°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
	0	0	0.018 94.449 1.161	0.007 37.758 0.463	
Left 0	0	0	0.125 454.326 4.672	0.071 256.500 2.638	Right 0
	0	0	0.303 703.010 6.059	0.153 355.363 3.067	
S & L 0		Short 3.951 2685.392 471.939			S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.05°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
	0	0	0	0.025 132.207 1.624	
Left 0	0	0	0	0.195 707.911 7.283	Right 0
	0	0	0	0.456 1060.301 9.145	
S & L 0		Short 3.951 2686.379 471.947			S & R 0

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.075°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0	
	0	0	0	0.020 108.000 1.328	
Left 0	0	0	0	0.147 533.614 5.484	Right 0.136 393.908 3.785
	0	0	0	0.372 864.897 7.456	
S & L 0		Short 3.789 2477.783 470.465			S & R 0.163 208.596 1.483

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.10°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
	0	0	0	0	
Left 0	0	0	0	0	Right 0.673 1896.694 18.023
	0	0	0	0	
S & L 0		Short 0			S & R 3.953 2690.105 471.977

Vertical Launch Angle = 30.2°
Off-Target Launch Angle = 0.15°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0.039 180.339 2.079	0.039 180.339 2.079	0
0	0.141 427.753 4.087	0.141 427.753 4.087	0
0	0.241 488.624 4.007	0.241 488.624 4.007	0
0	0.270 375.905 2.711	0.270 375.905 2.711	0

S & L 0 | Short 3.246 1641.556 610.659 | S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.0°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0 | Past 0 | P & R 0

Left 0 | Right 0

0	0	0.077 360.679 4.158	0
0	0	0.281 855.505 8.173	0
0	0	0.483 977.248 8.013	0
0	0	0.539 751 5.418	0

S & L 0 | Short 3.246 1642.134 610.662 | S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.025°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0.051 239.215 2.773	0.027 121.463 1.385	
Left 0	0	0	0.184 556.680 5.307	0.098 298.826 2.866	Right 0
	0	0	0.326 657.157 5.380	0.157 320.091 2.634	
	0	0	0.396 552.605 3.989	0.144 198.627 1.429	
S & L 0		Short 3.246 1642.134 610.662			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.05°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0.077 360.679 4.158	
Left 0	0	0	0	0.281 855.505 8.173	Right 0
	0	0	0	0.482 976.432 8.007	
	0	0	0	0.540 752.049 5.424	
S & L 0		Short 3.246 1642.134 610.662			S & R 0

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.075°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX B.

RESULTS OF SIMULATION WITH VARIOUS LAUNCH ANGLES

P & L 0		Past 0			P & R 0
	0	0	0	0.060 282.068 3.259	
	0	0	0	0.217 656.479 6.254	
Left 0	0	0	0	0.393 795.002 6.516	Right 0.236 548.573 4.954
	0	0	0	0.471 658.385 4.753	
S & L 0		Short 3.206 1602.535 610.074			S & R 0.044 43.755 0.331

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.10°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

P & L 0		Past 0			P & R 0
	0	0	0	0	
	0	0	0	0	
Left 0	0	0	0	0	Right 1.376 2940.508 25.736
	0	0	0	0	
S & L 0		Short 0			S & R 3.250 1646.291 610.405

Vertical Launch Angle = 30.3°
Off-Target Launch Angle = 0.15°
Intercept Range = 4.0 km

Each Block Contains: # of Fragments
Total Mass
Total KE

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
DEFINITION MODULE NewFrag;

FROM NewTBM IMPORT G, PositionType;

CONST C = 1.0;

TYPE      TargetType = RECORD
Rows: INTEGER;
Columns: INTEGER;
 END RECORD;

IntArrayType = ARRAY INTEGER, INTEGER OF INTEGER;
RealArrayType = ARRAY INTEGER, INTEGER OF REAL;

FragObj = OBJECT
Mass : REAL;
Area : REAL;
OriginalArea: REAL;
Velocity : PositionType;
Position : PositionType;
CurrentVel: REAL;
Damage : REAL;
LengthofBlock: REAL;
 SouthWest: PositionType;
DX: REAL;
DZ: REAL;
TGT: TargetType;

ASK METHOD ObjInit;
ASK METHOD GetMass(IN M: REAL);
ASK METHOD CalcArea;
ASK METHOD CalcVelocity(IN V: PositionType);
ASK METHOD GetPosition(IN P: PositionType);
ASK METHOD SetTargetRegion(IN LE: REAL; IN WI: REAL; IN RO:  INTEGER; IN CO: INTEGER; IN DI:
                                    REAL);
ASK METHOD CalcDamage;
ASK METHOD Travel;
END OBJECT;

FragmentsType = ARRAY INTEGER OF FragObj;

VAR
Fragment : FragObj;
Fragments : FragmentsType;
FragCount : IntArrayType;
MassTotal :  RealArrayType;
KETotal : RealArrayType;
END MODULE.
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
DEFINITION MODULE NewRunRep;

PROCEDURE RunReplications;

END MODULE.


DEFINITION MODULE NewTBM;

FROM RandMod IMPORT RandomObj;
FROM NewFrag IMPORT IntArrayType, RealArrayType;

CONST  G = .0098;
Accel = .023875;
Area = .6082;
WM = 7000.0;
MBT = 90.0;
seed1 = 1;
seed2 = 3;
seed3 = 5;
seed4 = 7;
C1 = 1.0;

TYPE PositionType = RECORD
X: REAL;
Y: REAL;
Z: REAL;
END RECORD;

MissileObj = OBJECT
Dist:  REAL;
WarheadMass:  REAL;
LA:  REAL;
OA:  REAL;
TBMPosition:  PositionType;
TBMVelocity:  PositionType;
Acceleration:  REAL;
Pieces:  INTEGER;
MotorBurnTime:  REAL;
MissileArea:  REAL;
InterceptRange:  REAL;
AfterMotorSpeed:  REAL;
InterceptTime:  REAL;
Altitude:  REAL;
MeanFragments:  REAL;
Length:  REAL;
Width: REAL;
Rows: INTEGER;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
Columns: INTEGER;
TotalFragCount: IntArrayType;
TotalMD: RealArrayType;
TotalKE: RealArrayType;
TPLD: REAL;
TPLMD: REAL;
TPLKE: REAL;
TSLD: REAL;
TSLMD: REAL;
TSLKE: REAL;
TLD: REAL;
TLMD: REAL;
TLKE: REAL;
TPRD: REAL;
TPRMD: REAL;
TPRKE: REAL;
TSRD: REAL;
TSRMD: REAL;
TSRKE: REAL;
TRD: REAL;
TRMD: REAL;
TRKE: REAL;
TPD: REAL;
TPMD: REAL;
TPKE: REAL;
TSD: REAL;
TSMD: REAL;
TSKE: REAL;

ASK METHOD ObjInit;
ASK METHOD FlyTraj;
ASK METHOD BreakUp;
ASK METHOD GetInterceptRange(IN R: REAL);
ASK METHOD GetMeanFragment(IN MF: REAL);
ASK METHOD GetAngles(IN VA: REAL; IN OTA: REAL);
ASK METHOD GetAltitude(IN ALT: REAL);
ASK METHOD GetTargetDimension(IN L: REAL; IN W: REAL;  IN R: INTEGER; IN C: INTEGER; IN D:
REAL); ASK METHOD ResetDamage(IN Zero: REAL);
ASK METHOD GetDamage(IN FC: INTEGER; IN MT: REAL;  IN KET: REAL; IN i: INTEGER; IN j:
INTEGER);   ASK METHOD GetPastLeftDamage(IN DD:REAL; IN DMD:REAL;  IN DKE:REAL);

ASK METHOD GetShortLeftDamage(IN DD:REAL; IN DMD:REAL;  IN DKE:REAL);
ASK METHOD GetLeftDamage(IN DD:REAL; IN DMD:REAL;  IN DKE:REAL);
ASK METHOD GetPastRightDamage(IN DD:REAL; IN DMD:REAL;  IN DKE:REAL);
ASK METHOD GetShortRightDamage(IN DD:REAL; IN DMD:REAL;  IN DKE:REAL);
ASK METHOD GetRightDamage(IN DD:REAL; IN DMD:REAL; IN DKE:REAL);
 ASK METHOD GetPastDamage(IN DD:REAL; IN DMD:REAL;  IN DKE:REAL);
ASK METHOD GetShortDamage(IN DD:REAL; IN DMD:REAL;  IN DKE:REAL);
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
END OBJECT;
VAR TBM: MissileObj;
P,R,S,T: RandomObj;
END MODULE.


DEFINITION MODULE NewWriteLine;

PROCEDURE WriteLine(IN String: STRING);

END MODULE.


IMPLEMENTATION MODULE NewFrag;

FROM RandMod IMPORT RandomObj;
FROM NewTBM IMPORT PositionType, G, TBM,R,S,T;
FROM MathMod IMPORT SQRT,POWER, pi,CEIL;
FROM NewWriteLine IMPORT WriteLine;

OBJECT FragObj;
{...............................................................}
ASK METHOD ObjInit;
{...............................................................}
BEGIN
NEW(Position);
NEW(Velocity);
NEW(TGT);
NEW(SouthWest);
END METHOD;

{...............................................................}
ASK METHOD GetMass(IN M: REAL);
{...............................................................}
BEGIN
Mass := M;
END METHOD;

{...............................................................}
ASK METHOD CalcArea;
{...............................................................}
VAR
Radius: REAL;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
BEGIN
Radius := ((3.0/(4.0*pi)) * (0.001216425 * Mass));
Area := pi * POWER(Radius,(2.0/3.0));
OriginalArea := Area;
END METHOD;

{...................................................................}
ASK METHOD CalcVelocity(IN V: PositionType);
{...................................................................}
VAR
kickx,kicky,kickz : REAL;

BEGIN
kickx := (ASK T TO Normal(0.0,0.01));
kicky := (ASK T TO Normal(0.0,0.01));
kickz := (ASK T TO Normal(0.0,0.01));
Velocity.X := V.X + kickx;
Velocity.Y := V.Y + kicky;
Velocity.Z := V.Z + kickz;
END METHOD;

{...................................................................}
ASK METHOD GetPosition(IN P: PositionType);
{...................................................................}
BEGIN
Position.X := P.X;
Position.Y := P.Y;
Position.Z := P.Z;
END METHOD;

{...................................................................}
ASK METHOD SetTargetRegion(IN LE: REAL; IN WI: REAL; IN RO: INTEGER;   IN CO: INTEGER; IN DI:
                                                    REAL);
{...................................................................}
BEGIN
DX := LE/FLOAT(RO);
DZ := WI/FLOAT(CO);
SouthWest.X := DI - (0.5*FLOAT(RO)*DX);
SouthWest.Z := 0.0 - (0.5*FLOAT(CO)*DZ);
TGT.Rows := RO;
TGT.Columns := CO;
NEW(FragCount, 1..RO, 1..CO);
NEW(MassTotal, 1..RO, 1..CO);
NEW(KETotal, 1..RO, 1..CO);
END METHOD;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
{..................................................................}
ASK METHOD CalcDamage;
{..................................................................}
VAR
I,J, PLD,PD,PRD,SLD,SD,SRD,LD,RD: INTEGER;
PLMD,PLKE,PMD,PKE,PRMD,PRKE,SLMD,SLKE,SMD,SKE, SRMD,SRKE,LMD,LKE,RMD,RKE: REAL;

BEGIN
I := CEIL((Position.X - SouthWest.X)/DX);
J := CEIL((Position.Z - SouthWest.Z)/DZ);
IF (I > 0) AND (I <= TGT.Rows)
IF (J > 0) AND (J <= TGT.Columns)
INC(FragCount[I,J]);
MassTotal[I,J] := MassTotal[I,J] + Mass;
KETotal[I,J] := KETotal[I,J] + (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetDamage(FragCount[I,J],MassTotal[I,J],KETotal[I,J],I,J);
END IF;
END IF;
IF I > TGT.Rows
IF J <= 0
PLD := 1;
PLMD := Mass;
PLKE := (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetPastLeftDamage(FLOAT(PLD),PLMD,PLKE);
END IF;
IF (J > 0) AND (J<= TGT.Columns)
PD := 1;
PMD := Mass;
PKE := (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetPastDamage(FLOAT(PD),PMD,PKE);
END IF;
IF J > TGT.Columns
PRD := 1;
PRMD := Mass;
PRKE := (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetPastRightDamage(FLOAT(PRD),PRMD,PRKE);
END IF;
END IF;
IF I <= 0
IF J <= 0
SLD := 1;
SLMD := Mass;
SLKE := (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetShortLeftDamage(FLOAT(SLD),SLMD,SLKE);
END IF;
IF (J > 0) AND (J <= TGT.Columns)
SD := 1;
SMD := Mass;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
SKE := (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetShortDamage(FLOAT(SD),SMD,SKE);
END IF;
IF J > TGT.Columns
SRD := 1;
SRMD := Mass;
SRKE := (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetShortRightDamage(FLOAT(SRD),SRMD,SRKE);
END IF;
END IF;
IF (I > 0) AND (I <= TGT.Rows)
IF J <= 0
LD := 1;
LMD := Mass;
LKE := (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetLeftDamage(FLOAT(LD),LMD,LKE);
END IF;
IF J >= TGT.Columns
RD := 1;
RMD := Mass;
RKE := (0.5*Mass*POWER(CurrentVel,2.0));
ASK TBM TO GetRightDamage(FLOAT(RD),RMD,RKE);
END IF;
END IF;
END METHOD;

{..................................................................}
ASK METHOD Travel;
{..................................................................}
VAR
t, rho,VTerm : REAL;

BEGIN
t := 0.0;
WHILE (Position.Y > 0.0) AND (Velocity.X > 0.0)
CASE ROUND(Position.Y)
WHEN 0..1:
rho := 1.1117;
WHEN 2:
rho := 1.0066;
WHEN 3:
rho := 0.90925;
WHEN 4:
rho := 0.81935;
WHEN 5:
rho := 0.73643;
WHEN 6:
rho := 0.66011;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
WHEN 7:
rho := 0.59002;
WHEN 8:
rho := 0.52579;
WHEN 9:
rho := 0.46706;
WHEN 10:
rho := 0.41351;
WHEN 11:
rho := 0.3648;
WHEN 12:
rho := 0.31194;
WHEN 13:
rho := 0.2666;
WHEN 14:
rho := 0.22786;
WHEN 15:
rho := 0.19475;
WHEN 16:
rho := 0.16647;
WHEN 17:
rho := 0.1423;
WHEN 18:
rho := 0.12165;
WHEN 19:
rho := 0.104;
WHEN 20:
rho := 0.08991;
WHEN 21..25:
rho := 0.04;
WHEN 26..30:
rho := 0.01841;
WHEN 31..40:
rho := 0.0039957;
WHEN 41..50:
rho := 0.0010269;
WHEN 51..60:
rho := 0.00030592;
WHEN 61..70:
 rho := 0.000087535;
WHEN 71..80:
rho := 0.00001999;
WHEN 81..150:
rho := 0.0000004974;
END CASE;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
CurrentVel := SQRT(POWER(Velocity.X,2.0) + POWER(Velocity.Y,2.0) + POWER(Velocity.Z,2.0));
Velocity.X := ((Velocity.X*1000.0) - ((CurrentVel*1000.0*Velocity.X*1000.0*rho*Area*C)/(2.0*Mass)))
        /1000.0;
Velocity.Y := ((Velocity.Y*1000.0) - (G*1000.0) - ((CurrentVel*1000.0*Velocity.Y*1000.0*rho*Area*C)/
        (2.0*Mass)))/1000.0;
Velocity.Z := ((Velocity.Z*1000.0) - ((CurrentVel*1000.0*Velocity.Z*1000.0*rho*Area*C)/(2.0*Mass)))
        /1000.0;
Position.X := Position.X + (Velocity.X);
Position.Y := Position.Y + (Velocity.Y);
Position.Z := Position.Z + (Velocity.Z);
Area := ASK S TO Normal(OriginalArea, 0.05);
t := t + 1.0;
END WHILE;

ASK SELF TO CalcDamage;
END METHOD;
END OBJECT;
END MODULE.


IMPLEMENTATION MODULE NewRunRep;

FROM NewTBM IMPORT TBM;
FROM SimMod IMPORT StartSimulation,ResetSimTime;
FROM NewWriteLine IMPORT WriteLine;
FROM MathMod IMPORT POWER, SQRT;
FROM NewFrag IMPORT RealArrayType;

PROCEDURE RunReplications;
VAR
Z,p,q,c,d,ROW,COL: INTEGER;
APLD,APLMD,APLKE,ASLD, ASLMD,ASLKE,ALD,ALMD,ALKE,APRD,APRMD,APRKE,ASRD: REAL;
ASRMD,ASRKE,ARD,ARMD,ARKE,APD,APMD,APKE,ASD,ASMD,ASKE,PLD2,PLMD2,PLKE2: REAL;
SLD2,SLMD2 SLKE2,LD2,LMD2,LKE2,PRD2,PRMD2,PRKE2,SRD2,SRMD2,SRKE2,RD2: REAL;
RMD2,RKE2,PD2,PMD2,PKE2,SD2,SMD2,SKE2: REAL;
K,T,Range,VertAngle,OTAngle,MeanFragment,Len,Wid,Distance: REAL;
AvgFragCount,AvgMass,AvgKE,SQFragCount,SQMass,SQKE: RealArrayType;

BEGIN
NEW(TBM);
UTPUT("What is the downrange intercept range from the target?");
INPUT(Range);
OUTPUT("What is the desired vertical launch angle?");
INPUT(VertAngle);
OUTPUT("What is the desired off-target launch angle?");
INPUT(OTAngle);
OUTPUT("What is the mean number of fragments desired?");
INPUT(MeanFragment);
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
OUTPUT("What is the length of the target in KM?");
INPUT(Len);
OUTPUT("What is the width of the target in KM?");
INPUT(Wid);
OUTPUT("How many rows of blocks in the target region?");
INPUT(ROW);
OUTPUT("How many columns of blocks in the target region?");
INPUT(COL);
OUTPUT("What is the distance to the aimpoint in the center of the target?");
INPUT(Distance);
ASK TBM TO GetMeanFragment(MeanFragment);
ASK TBM TO GetAngles(VertAngle,OTAngle);
ASK TBM TO GetTargetDimension(Len,Wid,ROW,COL,Distance);
ASK TBM TO GetInterceptRange(Range);
NEW(AvgFragCount, 1..ROW, 1..COL);
NEW(AvgMass, 1..ROW, 1..COL);
NEW(AvgKE, 1..ROW, 1..COL);
NEW(SQFragCount, 1..ROW, 1..COL);
NEW(SQMass, 1..ROW, 1..COL);
NEW(SQKE, 1..ROW, 1..COL);
Z := 1;
WHILE Z < 101
ASK TBM TO FlyTraj;
StartSimulation;
FOR c := 1 TO ROW
FOR d := 1 TO COL
AvgFragCount[c,d] := AvgFragCount[c,d] + FLOAT(ASK TBM TotalFragCount[c,d]);
AvgMass[c,d] := AvgMass[c,d] + (ASK TBM TotalMD[c,d]);
AvgKE[c,d] := AvgKE[c,d] + (ASK TBM TotalKE[c,d]);
SQFragCount[c,d] := SQFragCount[c,d] + POWER(FLOAT(ASK TBM TotalFragCount[c,d]),2.0);
SQMass[c,d] := SQMass[c,d] + POWER(ASK TBM TotalMD[c,d],2.0);
SQKE[c,d] := SQKE[c,d] + POWER(ASK TBM TotalKE[c,d],2.0);
END FOR;
END FOR;
APLD := APLD + (ASK TBM TPLD);
APLMD := APLMD + (ASK TBM TPLMD);
APLKE := APLKE + (ASK TBM TPLKE);
ASLD := ASLD + (ASK TBM TSLD);
ASLMD := ASLMD + (ASK TBM TSLMD);
ASLKE := ASLKE + (ASK TBM TSLKE);
ALD := ALD + (ASK TBM TLD);
ALMD := ALMD + (ASK TBM TLMD);
ALKE := ALKE + (ASK TBM TLKE);
APRD := APRD + (ASK TBM TPRD);
APRMD := APRMD + (ASK TBM TPRMD);
APRKE := APRKE + (ASK TBM TPRKE);
ASRD := ASRD + (ASK TBM TSRD);
ASRMD := ASRMD + (ASK TBM TSRMD);
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
ASRKE := ASRKE + (ASK TBM TSRKE);
ARD := ARD + (ASK TBM TRD);
ARMD := ARMD + (ASK TBM TRMD);
ARKE := ARKE + (ASK TBM TRKE);
APD := APD + (ASK TBM TPD);
APMD := APMD + (ASK TBM TPMD);
APKE := APKE + (ASK TBM TPKE);
ASD := ASD + (ASK TBM TSD);
ASMD := ASMD + (ASK TBM TSMD);
ASKE := ASKE + (ASK TBM TSKE);
PLD2 := PLD2 + POWER(ASK TBM TPLD,2.0);
PLMD2 := PLMD2 + POWER(ASK TBM TPLMD,2.0);
PLKE2 := PLKE2 + POWER(ASK TBM TPLKE,2.0);
SLD2 := SLD2 + POWER(ASK TBM TSLD,2.0);
SLMD2 := SLMD2 + POWER(ASK TBM TSLMD,2.0);
SLKE2 := SLKE2 + POWER(ASK TBM TSLKE,2.0);
LD2 := LD2 + POWER(ASK TBM TLD,2.0);
LMD2 := LMD2 + POWER(ASK TBM TLMD,2.0);
LKE2 := LKE2 + POWER(ASK TBM TLKE,2.0);
PRD2 := PRD2 + POWER(ASK TBM TPRD,2.0);
PRMD2 := PRMD2 + POWER(ASK TBM TPRMD,2.0);
PRKE2 := PRKE2 + POWER(ASK TBM TPRKE,2.0);
SRD2 := SRD2 + POWER(ASK TBM TSRD,2.0);
SRMD2 := SRMD2 + POWER(ASK TBM TSRMD,2.0);
SRKE2 := SRKE2 + POWER(ASK TBM TSRKE,2.0);
RD2 := RD2 + POWER(ASK TBM TRD,2.0);
RMD2 := RMD2 + POWER(ASK TBM TRMD,2.0);
RKE2 := RKE2 + POWER(ASK TBM TRKE,2.0);
PD2 := PD2 + POWER(ASK TBM TPD,2.0);
PMD2 := PMD2 + POWER(ASK TBM TPMD,2.0);
PKE2 := PKE2 + POWER(ASK TBM TPKE,2.0);
SD2 := SD2 + POWER(ASK TBM TSD,2.0);
SMD2 := SMD2 + POWER(ASK TBM TSMD,2.0);
SKE2 := SKE2 + POWER(ASK TBM TSKE,2.0);
ASK TBM TO ResetDamage(0.0);
ResetSimTime(0.0);      Z := Z + 1;
END WHILE;
K := FLOAT(Z-1);
T := K - 1.0;
WriteLine(" ");
WriteLine("Intercept Distance is: " + REALTOSTR(Range) + " km");
WriteLine("Vertical Launch Angle is: " + REALTOSTR(VertAngle) + " degrees");
WriteLine("Off-Target Launch Angle is: " + REALTOSTR(OTAngle) + " degrees");
WriteLine("Intercept Altitude is:  " + REALTOSTR(ASK TBM Altitude) + " km");
WriteLine(" ");
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
FOR p := 1 TO ROW
FOR q := 1 TO COL
AvgFragCount[p,q] := AvgFragCount[p,q]/K;
AvgMass[p,q] := AvgMass[p,q]/K;
AvgKE[p,q] := AvgKE[p,q]/K;
SQFragCount[p,q] := SQRT(((SQFragCount[p,q]/T) - ((K/T)*POWER(AvgFragCount[p,q],2.0)))/K);
SQMass[p,q] := SQRT(((SQMass[p,q]/T) - ((K/T)*POWER(AvgMass[p,q],2.0)))/K);
SQKE[p,q] := SQRT(((SQKE[p,q]/T) - ((K/T)*POWER(AvgKE[p,q],2.0)))/K);
WriteLine("Block " + INTTOSTR(p) + "," + INTTOSTR(q) + " MEAN HITS: " +
REALTOSTR(AvgFragCount[p,q])              + " Std Error: " +  REALTOSTR(SQFragCount[p,q]));
WriteLine("Block " + INTTOSTR(p) + "," + INTTOSTR(q) + " MEAN MASS: "  + REALTOSTR(AvgMass[p,q])
          + "Std Error: " +  REALTOSTR(SQMass[p,q]));
WriteLine("Block " + INTTOSTR(p) + "," + INTTOSTR(q) + " MEAN KE: "  + REALTOSTR(AvgKE[p,q])
          + " Std Error: " + REALTOSTR(SQKE[p,q]));
END FOR;
END FOR;

APLD := APLD/K;
APLMD := APLMD/K;
APLKE := APLKE/K;
ASLD := ASLD/K;
ASLMD := ASLMD/K;
ASLKE := ASLKE/K;
ALD := ALD/K;
ALMD := ALMD/K;
ALKE := ALKE/K;
APRD := APRD/K;
APRMD := APRMD/K;
APRKE := APRKE/K;
ASRD := ASRD/K;
ASRMD := ASRMD/K;
ASRKE := ASRKE/K;
ARD := ARD/K;
ARMD := ARMD/K;
ARKE := ARKE/K;
APD := APD/K;
APMD := APMD/K;
APKE := APKE/K;
ASD := ASD/K;
ASMD := ASMD/K;
ASKE := ASKE/K;
PLD2 := SQRT((PLD2/T)-((K/T)*POWER(APLD,2.0)))/SQRT(K);
PLMD2 := SQRT((PLMD2/T)-((K/T)*POWER(APLMD,2.0)))/SQRT(K);
PLKE2 := SQRT((PLKE2/T)-((K/T)*POWER(APLKE,2.0)))/SQRT(K);
SLD2 := SQRT((SLD2/T)-((K/T)*POWER(ASLD,2.0)))/SQRT(K);
SLMD2 := SQRT((SLMD2/T)-((K/T)*POWER(ASLMD,2.0)))/SQRT(K);
SLKE2 := SQRT((SLKE2/T)-((K/T)*POWER(ASLKE,2.0)))/SQRT(K);
LD2 := SQRT((LD2/T)-((K/T)*POWER(ALD,2.0)))/SQRT(K);
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
LMD2 := SQRT((LMD2/T)-((K/T)*POWER(ALMD,2.0)))/SQRT(K);
LKE2 := SQRT((LKE2/T)-((K/T)*POWER(ALKE,2.0)))/SQRT(K);
PRD2 := SQRT((PRD2/T)-((K/T)*POWER(APRD,2.0)))/SQRT(K);
PRMD2 := SQRT((PRMD2/T)-((K/T)*POWER(APRMD,2.0)))/SQRT(K);
PRKE2 := SQRT((PRKE2/T)-((K/T)*POWER(APRKE,2.0)))/SQRT(K);
SRD2 := SQRT((SRD2/T)-((K/T)*POWER(ASRD,2.0)))/SQRT(K);
SRMD2 := SQRT((SRMD2/T)-((K/T)*POWER(ASRMD,2.0)))/SQRT(K);
SRKE2 := SQRT((SRKE2/T)-((K/T)*POWER(ASRKE,2.0)))/SQRT(K);
RD2 := SQRT((RD2/T)-((K/T)*POWER(ARD,2.0)))/SQRT(K);
RMD2 := SQRT((RMD2/T)-((K/T)*POWER(ARMD,2.0)))/SQRT(K);
RKE2 := SQRT((RKE2/T)-((K/T)*POWER(ARKE,2.0)))/SQRT(K);
PD2 := SQRT((PD2/T)-((K/T)*POWER(APD,2.0)))/SQRT(K);
PMD2 := SQRT((PMD2/T)-((K/T)*POWER(APMD,2.0)))/SQRT(K);
PKE2 := SQRT((PKE2/T)-((K/T)*POWER(APKE,2.0)))/SQRT(K);
SD2 := SQRT((SD2/T)-((K/T)*POWER(ASD,2.0)))/SQRT(K);
SMD2 := SQRT((SMD2/T)-((K/T)*POWER(ASMD,2.0)))/SQRT(K);
SKE2 := SQRT((SKE2/T)-((K/T)*POWER(ASKE,2.0)))/SQRT(K);

WriteLine("MISSES/LOCATIONS");
WriteLine("");
WriteLine("Mean PastLeft Misses:      " + REALTOSTR(APLD) + " Std Error: " +  REALTOSTR(PLD2));
WriteLine("Mean PastLeft Mass Damage: " + REALTOSTR(APLMD) + " Std Error: " +
REALTOSTR(PLMD2));
WriteLine("Mean PastLeft KE Damage:   " + REALTOSTR(APLKE) + " Std Error: " +  REALTOSTR(PLKE2));
WriteLine("Mean ShortLeft Misses:     " + REALTOSTR(ASLD) + " Std Error: " +  REALTOSTR(SLD2));
WriteLine("Mean ShortLeft Mass Damage:" + REALTOSTR(ASLMD) + " Std Error: "+REALTOSTR(SLMD2));
WriteLine("Mean ShortLeft KE Damage:  " + REALTOSTR(ASLKE) + " Std Error: "+ REALTOSTR(SLKE2));
WriteLine("Mean Left Misses:          " + REALTOSTR(ALD) + " Std Error: " + REALTOSTR(LD2));
WriteLine("Mean Left Mass Damage:     " + REALTOSTR(ALMD) + " Std Error: "+  REALTOSTR(LMD2));
WriteLine("Mean Left KE Damage:       " + REALTOSTR(ALKE) + " Std Error: " +  REALTOSTR(LKE2));
WriteLine("Mean PastRight Misses:     " + REALTOSTR(APRD) + " Std Error: " +  REALTOSTR(PRD2));
WriteLine("Mean PastRight Mass Damage:" + REALTOSTR(APRMD) + " Std Error: "+
REALTOSTR(PRMD2));  WriteLine("Mean PastRight KE Damage:  " + REALTOSTR(APRKE) + " Std Error: "
+ REALTOSTR(PRKE2));  WriteLine("Mean ShortRight Misses:    " + REALTOSTR(ASRD) + " Std Error: " +
REALTOSTR(SRD2));  WriteLine("Mean ShortRight Mass Damage:" + REALTOSTR(ASRMD) + " Std Error:
"+REALTOSTR(SRMD2));  WriteLine("Mean SkortRight KE Damage: " + REALTOSTR(ASRKE) + " Std Error:
" +  REALTOSTR(SRKE2));  WriteLine("Mean Right Misses:         " + REALTOSTR(ARD) + " Std Error: " +
REALTOSTR(RD2));  WriteLine("Mean Right Mass Damage:    " + REALTOSTR(ARMD) + " Std Error: " +
REALTOSTR(RMD2));  WriteLine("Mean Right KE Damage:      " + REALTOSTR(ARKE) + " Std Error: " +
REALTOSTR(RKE2));  WriteLine("Mean Past Misses:          " + REALTOSTR(APD) + " Std Error: " +
REALTOSTR(PD2));  WriteLine("Mean Past Mass Damage:     " + REALTOSTR(APMD) + " Std Error: " +
REALTOSTR(PMD2));  WriteLine("Mean Past KE Damage:       " + REALTOSTR(APKE) + " Std Error: " +
REALTOSTR(PKE2));  WriteLine("Mean Short Misses:         " + REALTOSTR(ASD) + " Std Error: " +
REALTOSTR(SD2));  WriteLine("Mean Short Mass Damage:    " + REALTOSTR(ASMD) + " Std Error: " +
REALTOSTR(SMD2));  WriteLine("Mean Short KE Damage:      " + REALTOSTR(ASKE) + " Std Error: " +
REALTOSTR(SKE2));
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
END PROCEDURE;

END MODULE.


IMPLEMENTATION MODULE NewTBM;

FROM NewWriteLine IMPORT WriteLine;
FROM SimMod IMPORT SimTime;
FROM RandMod IMPORT RandomObj,FetchSeed;
FROM NewFrag IMPORT FragObj, FragmentsType, IntArrayType, RealArrayType;
FROM MathMod IMPORT COS,SIN,SQRT,POWER, pi;

OBJECT MissileObj;

{..............................................................}
ASK METHOD ObjInit;
{..............................................................}
BEGIN
NEW(TBMVelocity);
NEW(TBMPosition);
NEW(P);
NEW(R);
NEW(S);
NEW(T);
ASK P TO SetSeed(FetchSeed(seed1));
ASK R TO SetSeed(FetchSeed(seed2));
ASK S TO SetSeed(FetchSeed(seed3));
ASK T TO SetSeed(FetchSeed(seed4));
Acceleration := Accel;    {km/sec2}
WarheadMass := WM;        {kg}
MissileArea := Area;      {m^2}
MotorBurnTime := MBT ;    {secs}
AfterMotorSpeed := Acceleration * MotorBurnTime;
END METHOD;

{..............................................................}
ASK METHOD FlyTraj;
{..............................................................}
VAR
InterceptPt, rho, CurrentVel: REAL;

BEGIN
NEW(TotalFragCount, 1..Rows, 1..Columns);
NEW(TotalMD, 1..Rows, 1..Columns);
NEW(TotalKE, 1..Rows, 1..Columns);
InterceptPt := Dist - InterceptRange;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
TBMPosition.X := (Accel*0.5*MBT*MBT*COS(LA)*COS(OA));
TBMPosition.Y := (Accel*0.5*MBT*MBT*SIN(LA));
TBMPosition.Z := (Accel*0.5*MBT*MBT*COS(LA)*SIN(OA));
TBMVelocity.X := (Accel*MBT)*COS(LA)*COS(OA);
TBMVelocity.Y := (Accel*MBT)*SIN(LA);
TBMVelocity.Z := (Accel*MBT)*COS(LA)*SIN(OA);
WarheadMass := 6000.0;
WHILE FLOAT(ROUND(TBMPosition.X)) < InterceptPt
CurrentVel := SQRT(POWER(TBMVelocity.X,2.0)+POWER(TBMVelocity.Y,2.0) +
              POWER(TBMVelocity.Z,2.0));
IF TBMPosition.Y > 0.0
CASE ROUND(TBMPosition.Y)
WHEN 0..1:
rho := 1.1117;
WHEN 2:
rho := 1.0066;
WHEN 3:
rho := 0.90925;
WHEN 4:
rho := 0.81935;
WHEN 5:
rho := 0.73643;
WHEN 6:
rho := 0.66011;
WHEN 7:
rho := 0.59002;
WHEN 8:
rho := 0.52579;
WHEN 9:
rho := 0.46706;
WHEN 10:
rho := 0.41351;
WHEN 11:
rho := 0.3648;
WHEN 12:
rho := 0.31194;
WHEN 13:
rho := 0.2666;
WHEN 14:
rho := 0.22786;
WHEN 15:
rho := 0.19475;
WHEN 16:
rho := 0.16647;
WHEN 17:
rho := 0.1423;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
WHEN 18:
rho := 0.12165;
WHEN 19:
rho := 0.104;
WHEN 20:
rho := 0.08991;
WHEN 21..25:
rho := 0.04;
WHEN 26..30:
rho := 0.01841;
WHEN 31..40:
rho := 0.0039957;
WHEN 41..50:
rho := 0.0010269;
WHEN 51..60:
rho := 0.00030592;
WHEN 61..70:
rho := 0.000087535;
WHEN 71..80:
rho := 0.00001999;
WHEN 81..200:
rho := 0.0000004974;
END CASE;

END IF;
TBMVelocity.X := ((TBMVelocity.X*1000.0) - ((CurrentVel*1000.0* TBMVelocity.X*1000.0*
                  rho*MissileArea*C1) /(2.0*WarheadMass)))/1000.0;
TBMVelocity.Y := ((TBMVelocity.Y*1000.0) - (0.0098*1000.0) - ((CurrentVel*1000.0*TBMVelocity.Y*
                  1000.0*rho*MissileArea*C1)/(2.0*WarheadMass)))/1000.0;
TBMVelocity.Z := ((TBMVelocity.Z*1000.0) - ((CurrentVel*1000.0*TBMVelocity.Z*1000.0* rho*
                   MissileArea* C1)/(2.0*WarheadMass)))/1000.0;
TBMPosition.X := TBMPosition.X + TBMVelocity.X;
TBMPosition.Y := TBMPosition.Y + TBMVelocity.Y;
TBMPosition.Z := TBMPosition.Z + TBMVelocity.Z;
END WHILE;
ASK SELF TO BreakUp;
ASK SELF TO GetAltitude(TBMPosition.Y);
END METHOD;

{..............................................................}
ASK METHOD BreakUp;
{..............................................................}
VAR
I.K: INTEGER;
m.TotalMass: REAL;
Fragment: FragObj;
Fragments: FragmentsType;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
BEGIN
TotalMass := 0.0;
K := ASK R TO Poisson(ASK SELF MeanFragments);
IF K <= 1
K := 2;
END IF;
NEW(Fragments, 1..5*K);
I := 1;
WHILE TotalMass < WarheadMass
m := (WarheadMass/FLOAT(K))*(ASK P TO Exponential(1.0));
TotalMass := TotalMass + m;
IF m > 70.0
IF TotalMass < WarheadMass
NEW(Fragments[I]);
ASK Fragments[I] TO GetMass(m);
ASK Fragments[I] TO CalcArea;
ASK Fragments[I] TO CalcVelocity(TBMVelocity);
ASK Fragments[I] TO GetPosition(TBMPosition);
ASK Fragments[I] TO SetTargetRegion(Length,Width,Rows,Columns, Dist);
ASK Fragments[I] TO Travel;
I := I + 1;
END IF;
END IF;
END WHILE;
END METHOD;

{..............................................................}
ASK METHOD GetInterceptRange(IN R: REAL);
{..............................................................}
BEGIN
nterceptRange := R;
END METHOD;

{..............................................................}
ASK METHOD GetMeanFragment(IN MF: REAL);
{..............................................................}
BEGIN
MeanFragments := MF;
END METHOD;

{..............................................................}
ASK METHOD GetAngles(IN VA: REAL; IN OTA: REAL);
{..............................................................}
BEGIN
LA := (VA*pi)/180.0;
OA := (OTA*pi)/180.0;
END METHOD;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
{................................................................}
ASK METHOD GetAltitude(IN ALT: REAL);
{................................................................}
BEGIN
Altitude := ALT;
END METHOD;

{................................................................}
ASK METHOD GetTargetDimension(IN L: REAL; IN W: REAL; IN R: INTEGER; IN C: INTEGER; IN D:
                                        REAL);
{................................................................}
BEGIN
Length := L;
Width := W;
Rows := R;
Columns := C;
Dist := D;
END METHOD;

{................................................................}
ASK METHOD GetDamage(IN FC: INTEGER; IN MT: REAL; IN KET: REAL;  IN i: INTEGER; IN j:
                                  INTEGER);
{................................................................}
BEGIN
TotalFragCount[i,j] := TotalFragCount[i,j] + FC;
TotalMD[i,j] := TotalMD[i,j] + MT;
TotalKE[i,j] := TotalKE[i,j] + KET;
END METHOD;

{................................................................}
ASK METHOD GetPastLeftDamage(IN DD: REAL; IN DMD: REAL; IN DKE: REAL);
{................................................................}
BEGIN
TPLD := TPLD + DD;
TPLMD := TPLMD + DMD;
TPLKE := TPLKE + DKE;
END METHOD;

{................................................................}
ASK METHOD GetShortLeftDamage(IN DD: REAL; IN DMD: REAL; IN DKE: REAL);
{................................................................}
BEGIN
TSLD := TSLD + DD;
TSLMD := TSLMD + DMD;
TSLKE := TSLKE + DKE;
END METHOD;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
{...............................................................}
ASK METHOD GetLeftDamage(IN DD: REAL; IN DMD: REAL; IN DKE: REAL);
{...............................................................}
BEGIN
TLD := TLD + DD;
TLMD := TLMD + DMD;
TLKE := TLKE + DKE;
END METHOD;

{...............................................................}
ASK METHOD GetPastRightDamage(IN DD: REAL; IN DMD: REAL; IN DKE: REAL);
{...............................................................}
BEGIN
TPRD := TPRD + DD;
TPRMD := TPRMD + DMD;
TPRKE := TPRKE + DKE;
END METHOD;

{...............................................................}
ASK METHOD GetShortRightDamage(IN DD: REAL; IN DMD: REAL; IN DKE: REAL);
{...............................................................}
BEGIN
TSRD := TSRD + DD;
TSRMD := TSRMD + DMD;
TSRKE := TSRKE + DKE;
END METHOD;

{...............................................................}
ASK METHOD GetRightDamage(IN DD: REAL; IN DMD: REAL; IN DKE: REAL);
{...............................................................}
BEGIN
TRD := TRD + DD;
TRMD := TRMD + DMD;
TRKE := TRKE + DKE;
END METHOD;

{...............................................................}
ASK METHOD GetPastDamage(IN DD: REAL; IN DMD: REAL; IN DKE: REAL);
{...............................................................}
BEGIN
TPD := TPD + DD;
TPMD := TPMD + DMD;
TPKE := TPKE + DKE;
END METHOD;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
{..............................................................}
ASK METHOD GetShortDamage(IN DD: REAL; IN DMD: REAL; IN DKE: REAL);
{..............................................................}
BEGIN
TSD := TSD + DD;
TSMD := TSMD + DMD;
TSKE := TSKE + DKE;
END METHOD;

{..............................................................}
ASK METHOD ResetDamage(IN Zero: REAL);
{..............................................................}
VAR
k,n: INTEGER;

BEGIN
FOR k := 1 TO Rows
FOR n := 1 TO Columns
TotalFragCount[k,n] := 0;
TotalMD[k,n] := Zero;
TotalKE[k,n] := Zero;
END FOR;
END FOR;

TPLD := Zero;
TPLMD := Zero;
TPLKE := Zero;
TSLD := Zero;
TSLMD := Zero;
TSLKE := Zero;
TLD := Zero;
TLMD := Zero;
TLKE := Zero;
TPRD := Zero;
TPRMD := Zero;
TPRKE := Zero;
TSRD := Zero;
TSRMD := Zero;
TSRKE := Zero;
TRD := Zero;
TRMD := Zero;
TRKE := Zero;
TPD := Zero;
TPMD := Zero;
TPKE := Zero;
TSD := Zero;
TSMD := Zero;
```

APPENDIX C.

MODSIM SIMULATION SOURCE CODE

```
TSKE := Zero;
END METHOD;
END OBJECT;

END MODULE.


IMPLEMENTATION MODULE NewWriteLine;

FROM IOMod IMPORT FileUseType(Output);
FROM IOMod IMPORT StreamObj;
FROM UtilMod IMPORT DateTime;
FROM SimMod IMPORT SimTime;
FROM NewTBM IMPORT TBM;

VAR
DT : STRING;
TraceStream : StreamObj;

PROCEDURE WriteLine(IN String: STRING);
BEGIN
IF (TraceStream = NILOBJ)
NEW(TraceStream);
ASK TraceStream TO Open("sim___.out", Output);
DateTime(DT);
ASK TraceStream TO WriteString(DT);
ASK TraceStream TO WriteLn;
ASK TraceStream TO WriteLn; E
ND IF;
ASK TraceStream TO WriteReal(SimTime(), 7,3);
ASK TraceStream TO WriteString(" -- " + String);
ASK TraceStream TO WriteLn;
END PROCEDURE;

PROCEDURE WriteLineClose;
BEGIN
ASK TraceStream TO Close;
END PROCEDURE;

END MODULE.
```

LIST OF REFERENCES

1. Fallon, William G., "Combating the Ballistic Missile Threat," *U.S. Naval Institute Proceedings*, vol 120/7/1097, pp 31-34, July 1994.

2. Lennox, D., *Jane's Missiles,* Iraqi Offensive Weapons, Jane's Information Group Limited. Sentinel House, 1994.

3. Herrman, E. E., *Exterior Ballistics,* p 45, U.S. Naval Institute, 1935.

4. *MATLAB High-Performance Numeric Computation and Visualization Software User's Guide*, The MathWorks,Inc., 1993.

5. Neuenfeldt, B. D. and Henderson, W. K., *A Survey of Uncontrolled Satellite Reentry and Impact Prediction.* Master's Thesis, Naval Postgraduate School, Monterey, California, September 1993.

6. Gray, D. E., *American Institue of Physics Handbook*, Third Edition, p 2-137, McGraw-Hill, Inc., 1982.

7. Halliday, D. and Resnick, R., *Fundamentals of Physics*, 3rd Edition, p E6-3, John Wiley & Sons, 1988.

8. Evans, E. D. and Mozzicato, R. F., Navy TBM Debris Characteriztion Review, *TBM Separation Characteristics* Massachusetts Institute of Technology Lincoln Laboratory, 26 July 1994.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, California 93943-5101	2
3.	Department of Operations Research Naval Postgraduate School Monterey, California 93943-5002	
	Attn: Chairman, OR/Pd	1
	Attn: Dr. Donald Gaver, OR/Gv	3
	Attn: Dr. Arnold Buss, OR/Bu	1
4.	Commander U.S. Army Missile Command AMSMI-RDAC Huntsville, Alabama 35898-5242	
	Attn: Dr B. W. Fowler	1
	Attn: Mr. Don Peterson	1
5.	Department of the Navy PEO (TAD)-B4 2531 Jefferson Davis Hwy Arlington, VA 22242-5170 CDR Roesner	1
6.	Lincoln Laboratory 244 Wood St Lexington, Massachusetts 02173 Attn: Dr. Eric Evans, Rm J149	1
7.	Commander Operational Test and Evaluation Force Norfolk, Virginia 23511 Attn: James B. Duff, Technical Director	1

8. Joseph M. Oliver 1
14044 Pimberton Dr
Hudson, Florida 34667