NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

ADAPTIVE DISCRETE EVENT SIMULATION FOR ANALYSIS OF HARPY SWARM ATTACK

by

Brandon J. Cobb

September 2011

Thesis Advisor: Arnold Buss
Second Reader: Jeffrey Kline

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704–0188*
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 Washington 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** September 2011	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE Adaptive Discrete Event Simulation for Analysis of Harpy Swarm Attack			**5. FUNDING NUMBERS**
6. AUTHOR(S) Brandon J. Cobb			
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) N/A			**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. IRB Protocol number _____N.A.___________.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

Harpy swarm attacks are a new type of threat designed for Suppression of Enemy Air Defenses. Research into combating Harpy swarm attacks has been conducted but the simulation software used to date, Naval Simulation System, is inadequate for future research. A new and mission-focused simulation tool is necessary in order to advance research in defensive tactics against Harpy and other unmanned aerial vehicle threats (UAV).

This research develops a simulation model for a Harpy swarm attack using Simkit to meet the need for a mission specific analytical tool. The base model consists of a user-defined Harpy patrol area and a ship traversing the area on a course and speed also defined by the user. A total of 16 parameters are defined and implemented. The model records the time any Harpy impacts the ship to provide data for the response variable, the number of Harpy hits on the ship.

Main effect and full factorial regressions were performed as well as a partition tree to determine which parameters had the most significance on the number of Harpies which hit the ship. These model characteristics and future enhancements will provide researchers the ability to assess alternative anti-UAV swarm tactics.

14. SUBJECT TERMS Discrete Event Simulation, Simkit, Harpy Swarm, Counter UAV, Air Defense			**15. NUMBER OF PAGES** 59
			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** UU

NSN 7540–01–280–5500

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

THIS PAGE INTENTIONALLY LEFT BLANK

ADAPTIVE DISCRETE EVENT SIMULATION FOR ANALYSIS OF HARPY SWARM ATTACK

Brandon J. Cobb
Lieutenant, United States Navy
B.S., University of Arizona, 2004

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 2011

Author: Brandon J. Cobb

Approved by: Arnold Buss
Thesis Advisor

Jeffrey E. Kline
Second Reader

Robert F. Dell
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Harpy swarm attacks are a new type of threat designed for Suppression of Enemy Air Defenses. Research into combating Harpy swarm attacks has been conducted but the simulation software used to date, Naval Simulation System, is inadequate for future research. A new and mission-focused simulation tool is necessary in order to advance research in defensive tactics against Harpy and other unmanned aerial vehicle threats (UAV).

This research develops a simulation model for a Harpy swarm attack using Simkit to meet the need for a mission specific analytical tool. The base model consists of a user-defined Harpy patrol area and a ship traversing the area on a course and speed also defined by the user. A total of 16 parameters are defined and implemented. The model records the time any Harpy impacts the ship to provide data for the response variable, the number of Harpy hits on the ship.

Main effect and full factorial regressions were performed as well as a partition tree to determine which parameters had the most significance on the number of Harpies which hit the ship. These model characteristics and future enhancements will provide researchers the ability to assess alternative anti-UAV swarm tactics.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ANOVA	Analysis of Variance
ASCM	Anti-Ship Cruise Missile
DES	Discrete Event Simulation
DOE	Design of Experiment
IAI	Israeli Aerospace Industries
NOLH	Nearly Orthogonal Latin Hypercube
NM	Nautical Mile
NSS	Naval Simulation System
PRC	People’s Republic of China
RCS	Radar Cross Section
SEAD	Suppression of Enemy Air Defenses
SoF	Summary of Fit
UAV	Unmanned Aerial Vehicle

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Research pertaining to the Harpy Unmanned Aerial Vehicle (UAV), a technology acquired by the People's Republic of China (PRC) from the Israeli Aerospace Industries (IAI), has been conducted at the classified level. The research conducted by LT Kaiser (Kaiser, 2008), LT Hafer (Hafer, 2010), and LT Taylor (Taylor, 2010) studied various aspects of the impact of a Harpy swarm attack, as well as methods to counter the attack. The research was all performed using Naval Simulation System (NSS), a maritime multi-mission discrete simulation model. Two issues with NSS identified by each researcher prevented obtaining more detailed results. The first was NSS' lack of realism when applied to actual defensive weapon employment. Only one weapon could be assessed at a time due to NSS' inability to apply conditional logic to employ the best weapon for a given case. The second issue identified was the need to use analogous aerial systems to represent the Harpy due to NSS not having an accurate representation of unmanned aerial vehicles. Due to these limitations, the need for a new simulation tool is necessary to continue research in how to combat and counteract the abilities of a Harpy swarm attack.

The goal of this research is to develop a basic Simkit Discrete Event Simulation (DES) model capable of being expanded beyond the current limitations of NSS in order to provide a platform for future Harpy research. By designing the DES model in Simkit the flexibility afforded the modeler allows for future research to be carried out in a manner not restricted by NSS. By not having the simulation based on a predefined system of available options, and thereby forcing the researcher to make analogy comparisons, the researcher is able to better test and simulate defense systems and tactics not anticipated when NSS was designed.

Analysis of the Simkit Harpy model was conducted using a Nearly Orthogonal Latin Hypercube (NOLH) design consisting of six parameters totaling 17 design points with 100 replications of each design point. A main effects regression, a full factorial regression, and a partition tree were performed. Results of these regressions indicate that based on an unclassified range of data for the adjusted parameters the most significant factors in determining the number of Harpy hits that a ship will take are the probability, P(hit), of a Harpy hitting the target, the Beam Width of the Harpy sensor, the ship's speed transiting the area, and the dive speed of the Harpy. This test analysis demonstrates the Harpy model's usefulness for further development and use as an analytical tool for tactics development.

ACKNOWLEDGMENTS

I would like to thank the Naval Postgraduate School (NPS) for providing me with the opportunity to obtain a master's degree in operations research.

I would also like thank Professor Arnold Buss and Senior Lecturer Jeff Kline for their support and guidance while conducting my thesis research. The flexibility both you offered when adapting to changes throughout the research process allowed me to continue forward.

Finally, I would like to thank the other students and faculty of the Operations Research (OR) curriculum, all of whom have made this a worthwhile experience and a great learning environment.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND AND MOTIVATION

The People's Republic of China (PRC) is modernizing its maritime capabilities by all means available, including acquiring technology from other countries. PRC maritime forces' modernization now presents the greatest threat to American naval forces. The acquisition of technology from sources external to the PRC has allowed it to reverse engineer and adapt equipment to its needs as a direct means of presenting an area denial threat to American and allied forces.

Previous research pertaining to one specific piece of technology, the Harpy UAV, acquired by the PRC from Israeli Aerospace Industries (IAI), has been conducted at the classified level, and as such, the results and conclusions of that research will not be discussed. The research conducted by LT Kaiser (Kaiser, 2008), LT Hafer (Hafer, 2010), and LT Taylor (Taylor, 2010) looked at various characteristics of a Harpy swarm attack as well as methods to counter the attack. The research was all performed using Naval Simulation System (NSS), a maritime multimission discrete simulation model. Two issues with NSS identified by each researcher prevented obtaining more detailed results. The first was NSS' lack of realism when applied to actual defensive weapon employment. Only one weapon could be assessed at a time due to NSS' inability to apply conditional logic to employ the best weapon for a given case. The second issue identified was the need to use analogous aerial systems to represent the Harpy due to NSS not having an accurate representation of unmanned aerial vehicles. Due to these limitations, the need for a new simulation tool is necessary to continue research in how to combat and counteract the abilities of a Harpy swarm attack.

B. PROBLEM STATEMENT

This research's focus is to develop a basic Simkit Discrete Event Simulation (DES) model capable of being expanded beyond the current limitations of NSS in order to provide a platform for future Harpy research. By designing the DES model in Simkit the flexibility afforded the modeler allows for future research to be carried out in a manner not restricted by NSS. By not having the simulation based on a predefined system of available options and thereby forcing the researcher to make analogy comparisons, the researcher is able to better simulate and assess defensive systems and tactics not anticipated when NSS was designed.

In addition, Simkit's Event Graphs design allows generating new modules or events to be incorporated and implemented first visually as an Event Graph and then added to the base Harpy simulation model. The combination of modular simulation components provides the flexibility, extensibility, and scalability necessary to generate DES models that can be easily modified while still maintaining the desired functionality of the original DES model. This functionality allows the analyst to quickly build and evaluate alternative defense systems and tactics.

II. THE HARPY THREAT

A. HARPY DESCRIPTION

The Harpy UAV is a delta winged, all-composite, drone designed to provide an autonomous, “fire-and-forget” Suppression of Enemy Air Defenses (SEAD) capability. The Harpy is designed with an extended forebody; wingtip-mounted fins and rudders; a 20.5kW two-cylinder, two-stroke pusher engine and four deployable side-force surfaces, see Figure 1. Target detection is via an onboard passive radar receiver and detonation occurs above the target using a proximity fuse (IHS *Jane’s*, 2010).

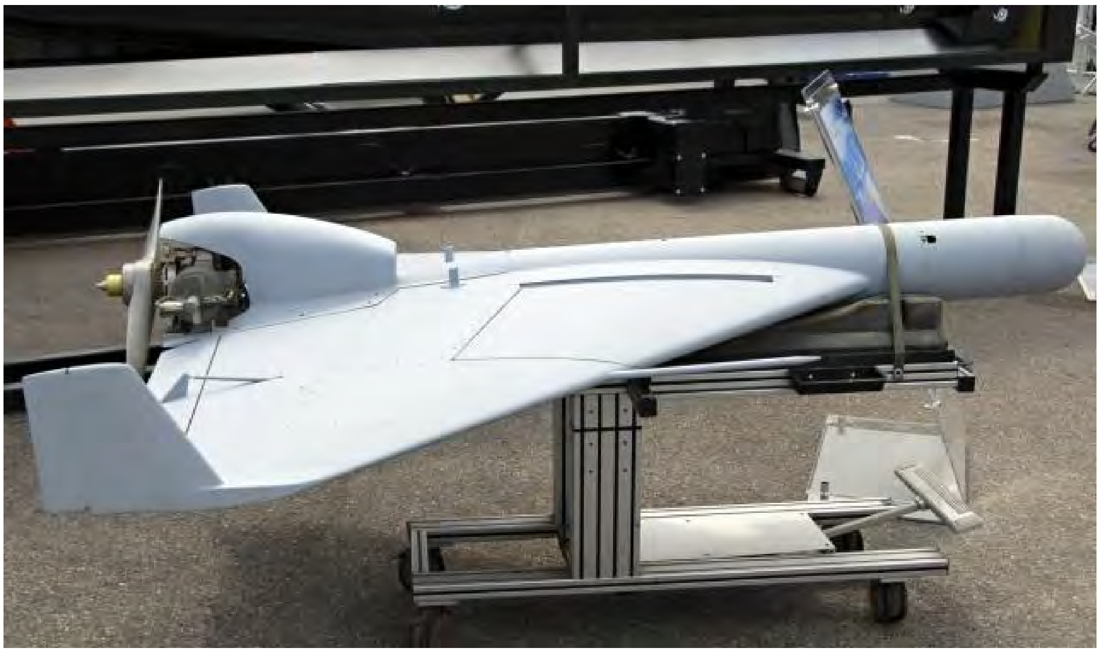


Figure 1. Harpy UAV (From IHS *Jane’s*, 2010)

The design and small size of a Harpy it produces a small Radar Cross Section (RCS). Combined with the lower cruising altitude and slower loiter speed a small RCS makes identifying a Harpy as a threat much more difficult. Combining these factors with the sheer number of contacts that can be detected by Aegis class ship’s radar means the potential for a Harpy contact to be designated as a nonthreat and not actively tracked becomes an issue that radar operators need to be aware of while operating in an area where a Harpy threat exists.

Harpy employment is via ground-based battery comprised of three launch units, a vehicle-mounted ground control shelter, a support vehicle and a trailer-mounted electrical power unit. Each launch vehicle contains nine launch/storage canisters, with two Harpy UAVs per canister, totaling 54 Harpy UAVs. Harpy UAVs can be fueled and defueled within their canisters and are launched via a rocket booster. Figure 2 is a Harpy UAV and launch vehicle (IHS *Jane's*, 2010).

Figure 2. Harpy UAV and Associated Launch Vehicle (From IHS *Jane's,* 2010)

B. HARPY MISSION PROFILE

Prior to launch, way points are programmed into each Harpy in order to provide navigation to the defined patrol area. Once the Harpy reaches the patrol area it begins a loitering pattern defined by pre-programmed way points. Upon detection of radar signal deemed a threat by the onboard prioritized threat library the Harpy begins to transit towards the radar signal source. As the Harpy approaches the radar source and optimal terminal dive angle is reached, the Harpy transitions into a terminal dive towards the radar source. If the radar source stops transmitting before the Harpy reaches the commit altitude the Harpy aborts the attack and returns to the preprogrammed loitering pattern. In the event that a Harpy does not find a target and fuel runs out the Harpy self-destructs (IHS *Jane's,* 2010). Figure 3 shows a Harpy during the terminal dive phase of the mission profile.

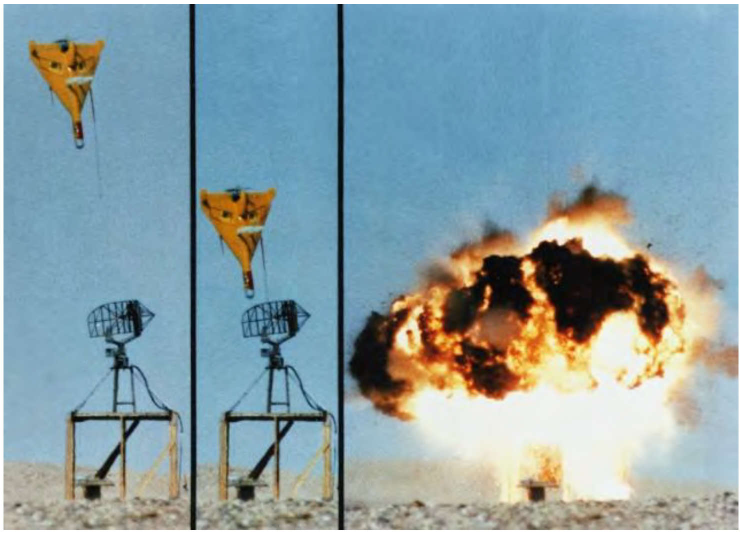


Figure 3. Harpy UAV Performing Terminal Dive on Radar Emitter (From IHS *Jane's*, 2010)

C. HARPY EMPLOYMENT

The Harpy UAV is designed to be employed as a group from all three launch units simultaneously. The 54 Harpy UAVs will loiter in their pre-programmed patrol area listening for any radar source designated as a threat. An individual Harpy has little chance of damaging a ship due to the likelihood of a ship's defensive weapons being able to eliminate it prior to impact. The danger presented by the Harpy is due to the number of Harpies present. A Harpy swarm, defined by multiple Harpies attacking at once, have the potential ability to either overwhelm a ship's defenses or to force the ship to expend enough weapons that not enough munitions are available for any subsequent attacks by either more Harpies or Anti-Ship Cruise Missiles (ASCM).

THIS PAGE INTENTIONALLY LEFT BLANK

III. MODELING AND SIMULATION

A. DISCRETE EVENT SIMULATION

Discrete Event Simulation (DES) describes an event-oriented simulation where events may happen at any time. The operation of the system is represented as a chronological sequence of events. Each event occurs at an instant in time and marks a change of state in the system.

Due to a DES relying on the triggering of events to progress, rather than a specified amount of time passing, “the timing of the occurrence of events is controlled by the Future Event List, which is nothing more than a “to-do” list of scheduled events. Whenever an event is scheduled to occur, an event notice is created and stored on the future event list. Every event notice contains two pieces of information: (1) what even is being scheduled; and (2) the (simulated) time at which the event is to occur” (Buss, 2001a). Therefore; a DES does not progress in a stepped time increment manner, allowing periods of time where events are not occurring to be skipped. This reduces the simulation run time when compared to an identical simulation proceeding incrementally in time.

DES models are comprised of four basic elements: state variables, events, parameters, and scheduling relationships between events. State variables are variables that have the potential of changing value during a simulation run. In the case of a queuing system, the state variable would be the number of people in a queue. The collection of all state variables is called the state space, which gives a complete description of the DES model at any point during the simulation.

Events are occurrences that have the potential to change the state of the system. The arrival of a new person in a queuing system is an example of an event which has the potential to change the state of the system. All events specify their state transition function and have an associated event time in order to be completely defined.

Parameters are the variables that do not change value during a simulation. Examples of parameters are the max and min number of servers available to service a queuing system.

Scheduling relationships between events are the rules that determine what the next event to occur will be. An example of a scheduling relationship is the starting of a second server in a queuing system once a specified number, based on a parameter, of customers are waiting in the queue for service. The combination of state variables, events, parameters, and scheduling relationships are expressed graphically in a format called Event Graphs. Buss defines event graphs as:

> Event Graphs are a way of representing the Future Event List logic for a discrete-event model. An Event Graph consists of nodes and directed edges. Each node corresponds to an event, or state transition, and each edge corresponds to the scheduling of other events. Each edge can optionally have an associated Boolean condition and/or a time delay. (2001a)

Figure 4 shows the fundamental construct for Event Graphs and is interpreted as follows: the occurrence of event A causes event B to be scheduled after a time delay of t, providing that condition (i) is true.

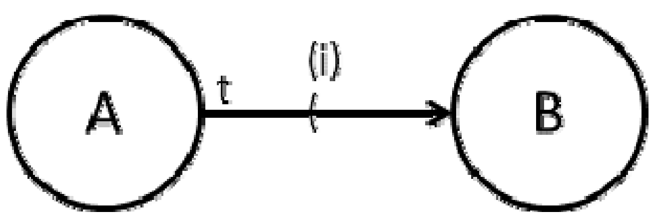


Figure 4. Basic Event Graph Construct (From Buss, 2001a)

Figure 4 represents the most basic construct of Event Graphs and allows for the creation of almost any DES. “In practice, however, there are two simple extensions that enhance event graph models’ ease of use and enable much simpler models to be created. These extensions are the cancelling edge and the ability to pass parameters on edges” (Buss, 2001a). Figure 5 represents a canceling edge and is interpreted as: “Whenever event A occurs, then if condition (i) is true, the first occurrence of event B is removed from the event list” (Buss, 2001a).

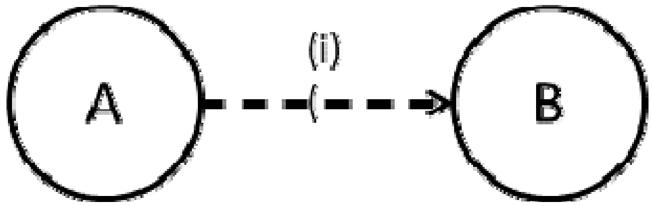


Figure 5. Cancelling Edge Event Graph (From Buss, 2001a)

The final extension, the ability to pass parameters on edges, is used as a "means of passing information about the current state of the model to a future event" (Buss, 2001a). Figure 6 represents an Event Graph with the ability to pass a parameter on it's edge and is interpreted as: "When event A occurs then, if condition (i) is true, event B is scheduled to occur after a delay of t time units; when B occurs, its parameter k will be set to the value given by the expression j" (Buss, 2001a).

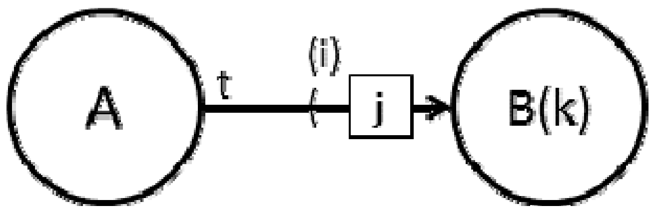


Figure 6. Event Graph with Parameter Passing (From Buss, 2001a)

With these basic components of Event Graphs, DES models are able to be generated using Simkit in order to represent complex simulations in a visually understandable way.

B. SIMKIT

Simkit is a Java software package, developed by Professor Arnold Buss at Naval Postgraduate School, used for implementing DES models. Simkit is structured so that an Event Graph can be translated almost directly into the various components of the Simkit templates necessary to model a DES.

The event and scheduling elements of a DES model are implemented in Simkit using a "user-defined 'do' method" (Buss, 2001b) representing each event. Scheduling elements are "executed using a method called 'waitDelay()' that has various signatures.

The simplest has signature (String, double), where the first argument is the name of the event without the 'do' and the second argument is the amount of simulated time between when the event is scheduled and when it occurs" (Buss, 2001b). The Simkit code necessary to implement the Basic Event Graph Construct of Figure 4 is implemented in Figure 7.

```
public void doA() {
     <code to perform state transition for event A>
     if (i) {
           waitDelay("B," t);
     }
}
```

Figure 7. Simkit Code Implementing Basic Event Graph Construct (From Buss, 2001b)

The other key component of Simkit which allows for highly flexible DES modeling are the two "'Listener' patterns to implement its component interoperability. The SimEventListener pattern is used to connect simulation components ... in a loosely coupled manner...the PropertyChangeListener pattern comes into play whenever a state variable changes value" (Buss, 2001b). The addition of Listeners allows for the building of large-scale, complex models effectively by creating small and manageable components and connecting them using the Listeners. Figure 8 demonstrates a SimEventListener relationship and the associated interpretation of the logic.

One simulation component shows interest in another's events by explicitly being registered as a SimEventListener to it. If there is a listener relationship (as in Figure 8), then whenever an Event from Source occurs, then after it has executed its state transitions and scheduled Events, the Event is sent to Listener. If Listener has an Event that is identical (in both name and signature) to the one it "hears" then it processes that Event as if it had scheduled it. The listening component does *not* re-dispatch heard Events to listeners, if it has any.

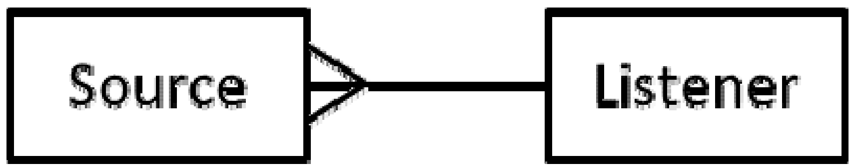


Figure 8. SimEventListener Relationship (From Buss, 2001b)

The combination of Simkit Events, Scheduling Edges, and Listeners allow for modular simulation components, thereby providing the flexibility, extensibility, and scalability necessary to generate DES models that can be easily modified while still maintaining the desired functionality of the original DES model.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. MODEL DEVELOPMENT

This research's goal is to develop a basic Simkit DES model capable of being expanded beyond the current limitations of NSS. The most important limitation of NSS, with regards to Harpy simulation testing, is the inability to use conditional logic when employing defensive weapons. Currently, NSS uses an "all or nothing" policy, which does not correlate to real world weapon employment. By designing the DES model in Simkit the flexibility afforded the modeler allows for future research in tactical decision making to be carried out in a manner not restricted by NSS. By not having the simulation based on a predefined system of available options, forcing the researcher to make analogy comparisons, the researcher is able to better test and simulate systems or tactics not anticipated when NSS was designed.

Due to the design intent of Simkit being based upon Event Graphs, generating new modules or events to be incorporated and implemented can be done first visually as an Event Graph and then added to the base Harpy simulation model. The following sections discuss the initial scenario built, the adjustable factors available, and key components of the model necessary for implementation.

A. SCENARIO

The scenario established for this research assumes that Harpies are being used as a defensive measure to prevent SPY-1D radar equipped ships from operating in a designated area. To accomplish the area denial, Harpies have been programmed to loiter in a predefined patrol box. Due to the Harpies having an unknown loitering pattern they are randomly distributed throughout the patrol box and move about randomly. With the Harpies patrolling in their defined area, a SPY-1D equipped ship begins to transit from east to west near the Harpy patrol box. Upon a Harpy detecting the ship the Harpy begins to transit towards the calculated intercept point of the ship, maintaining current altitude. Once the Harpy terminal dive conditions are satisfied, the Harpy performs the terminal dive with the intention of impacting the ship. The time at which any Harpy actually impacts the ship is recorded in order to determine the number of hits and the rate of hits.

B. ADJUSTABLE INPUT FACTORS

Table 1 is a list of currently implemented factors which can be adjusted prior to commencing a simulation run.

Ship Parameters	
Name	**Parameter Definition**
shipSpd	Speed at which ship travels (nm/hr)
shipInitX	Ship's initial x-coordinate (nm)*
shipInitY	Ship's initial y-coordinate (nm)*
shipFinX	Ship's final x-coordinate (nm)*
shipFinY	Ship's final y-coordinate (nm)*
Harpy Parameters	
patrolBoxXSize	X dimension of patrol box (nm)*
patrolBoxYSize	Y dimension of patrol box (nm)*
patrolBoxDist	Distance from origin to the center of patrol box (nm)*
harpyCruiseAlt	Harpy cruising altitude (ft)
harpyCruiseSpd	Harpy cruising speed (nm/hr)
harpyDiveSpd	Harpy dive speed (nm/hr)
harpyDiveAngle	Harpy dive angle from the horizon (degrees)
harpyDetRange	Harpy radar detection range (nm)
harpyBeamWidth	Harpy radar beam width (degrees)
harpyPhit	Harpy probability of hitting target
harpyQuant	Number of Harpies available
Other Parameters	
replications	Number of replications to perform

* Note: Coordinates are based on a grid with (0,0) as the center of the simulation area.

Table 1. Harpy Simkit Model List of Adjustable Parameters

C. MODEL COMPONENTS

1. Harpy Mover Manager

The HarpyMoverManager class controls the Harpy objects through each phase of the simulation. It contains all the subroutines associated with moving each Harpy as well as recording time a Harpy hits the target ship and incrementing the hit counter.

The Harpies patrol randomly, as previously discussed due to no known information about actual patrol patterns. If intelligence is available, defined patrol patterns can be created and employed. Defined patrol patterns could also be assigned to change during patrol phase as simulator or model conditions dictate.

The second phase of a Harpy objects movement occurs once the Harpy has detected the target. The StartAttack method controls the Harpy object during this phase of the simulation. The Harpy maintains the predefined altitude of Table 1, calculates the intercept point between the target and the Harpy and then proceeds towards intercept point. The intercept point is determined using the parameter values defined in Table 1 as well as the current location of both the Harpy and the target.

Phase three of a Harpy's movement is the terminal dive portion and is controlled by the StartDive method. The StartDive method initiates once the Harpy object has reached the appropriate distance from the ship based on the Harpy altitude and defined dive angle. Due to Simkit being constrained to movement in two dimensions an adjustment is needed in order to simulate the terminal dive portion of the Harpy object. The adjustment is performed by first determining the time necessary, in three dimensions, for the Harpy to traverse the distance from the start of the dive to the intercept point. Using the calculated time, the speed of the Harpy, as projected onto the X-Y plane, is then adjusted so that Harpy will impact the target at the correct time and location. Once the Harpy has intercepted the target, a uniform random number is generated and compared to the Harpy's probability of hitting the target (harpyPhit) from Table 1. If the random number is less than harpyPhit then the Harpy is recorded as a hit on the target and the total number of hits is incremented and the time of the hit is recorded, Figure 9 is the Event Graph representing the HarpyMoverManager and the associated phases of the simulation.

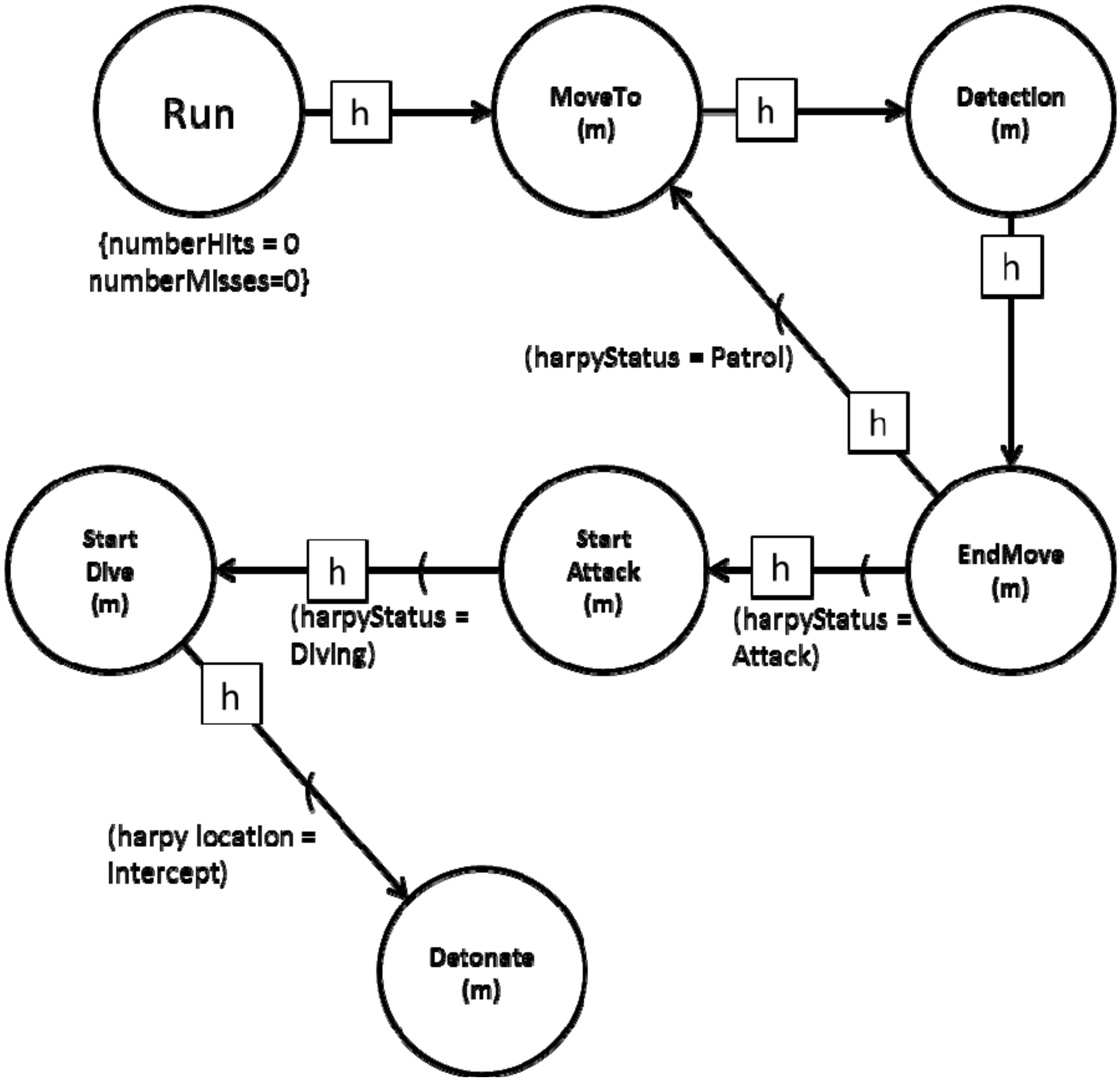


Figure 9. HarpyMoverManager Event Graph

2. Arc Cookie Cutter Mediator

Simkit contains a default class named CookieCutterSensor which serves the purpose of detecting other objects in a simulation, essentially acting as a radar. CookieCutterSensor can be defined to detect any object present or only those specifically designated. For the purposes of this research, each Harpy object has an attached

CookieCutterSensor and is designed to ignore other Harpy objects. By default, a CookieCutterSensor is a complete circle surrounding the object it is attached to and has a defined radius, harpyDetRange defines the radius of the sensor for the purposes of this research. In order to provide a range of options for representing the Harpy sensor the CookieCutterSensor class was modified using an ArcCookieCutterSensor class and an ArcCookieCutterMediator class. The ArcCookieCutterSensor class obtains the parameters harpyBeamWidth and harpyDetRange from the simulation allowing the default sensor to be modified by the ArcCookieCutterMediator.

ArcCookieCutterMediator converts the default 360-degree sensor into a sensor defined by arcs. The number and size of the arcs is determined by the harpyBeamWidth parameter. Due to current Java source code not allowing for the rotation and traversal of arcs to maintain a constant forward direction relative to the Harpy's movement the decision was made to represent the detection of the ship using a uniform random number. When the full 360 degree sensor detects the ship a uniform random number is drawn and compared to the ratio defined by the size of the harpyBeamWidth arc divided by the full circle. If the random number is less than the arc ratio then the detection is considered a real detection and the Harpy begins the startAttack phase of the simulation. The detection of the target by the ArcCookieCutterSensor is monitored by the series of Listeners illustrated in Figure 10.

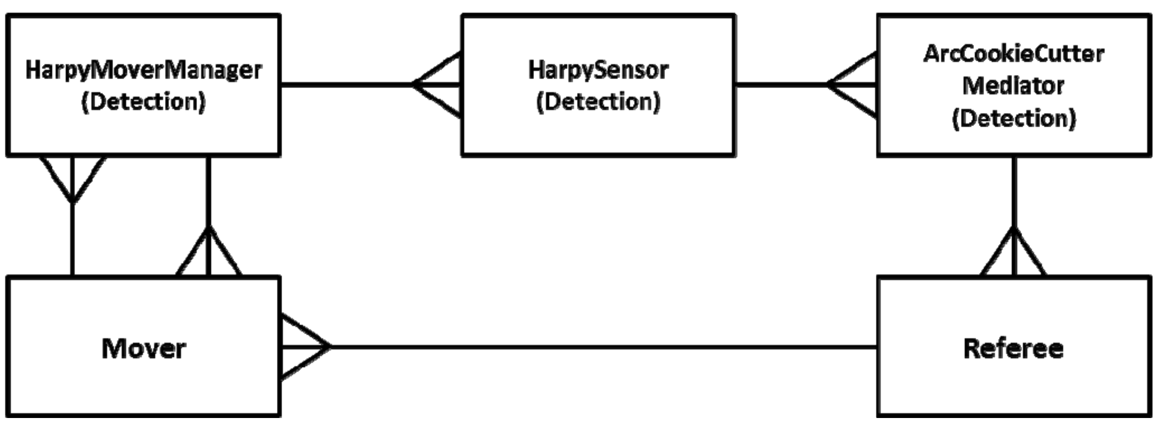


Figure 10. Listeners Associated with Harpy Detections

The rationale that allows for representing the detection as described is due to the movement of the Harpy object during the patrolling phase being random therefore, random movement having a random detection probability does not change the likelihood that a detection occurs. Future research using defined patrol patterns or waypoints for the Harpies will require redefining the ArcCookieCutterMediator in order to more accurately represent a Harpy sensor during the patrol phase.

3. Simple Harpy Adjudicator

The SimpleHarpyAdjudicator class determines whether the Harpy object actually hits the ship or is considered a miss. Upon completion of phase three, the terminal dive, of the HarpyMoverManager the Harpy "detonates" upon reaching the intercept point with the ship. SimpleHarpyAdjudicator contains a subroutine called "doDetonate" that then draws a uniform random number and compares it to the Harpy's probability of hitting the ship (harpyPhit). If the random number is less than harpyPhit then the Harpy is considered a "Hit" and the "doHit" subroutine is then called, otherwise the Harpy is a "miss" and the "doMiss" subroutine is called.

When called, the "doHit" subroutine obtains the current number of hits (numberHits) for the ship and increments it by one. When the "doMiss" subroutine is called the current number of misses (numberMisses) is incremented by one, see Figure 11. As currently implemented, the SimpleHarpyAdjudicator is the class in which the results of each simulation run are updated, i.e., the number of hits and misses, any other data which is desired to be recorded in future modifications of the DES model can be added to the SimpleHarpyAdjudicator class in order to have one centralized source for determining and assigning the parameters to be recorded.

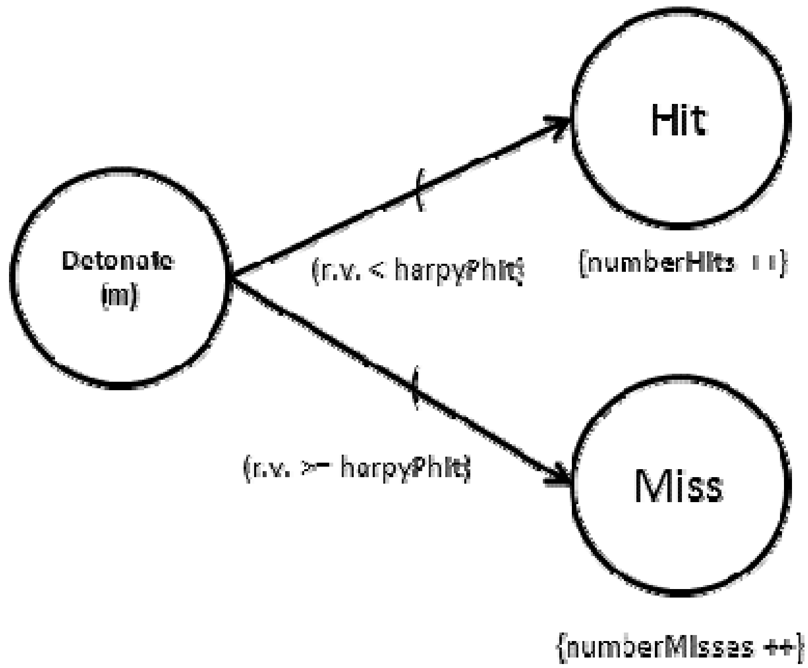


Figure 11. SimpleHarpyAdjudicator Event Graph

D. FUTURE DEFENSIVE WEAPON MODULES

As this research's focus is to develop a basic Simkit DES model capable of being expanded beyond the current limitations of NSS, defensive weapons are not incorporated into the model. However, a basic design for the implementation of defensive weapons is achievable through the use of Listeners assigned to the ship's sensor. An example of a defensive weapon module would incorporate a series of Listeners which implement a defensive weapon java class to use the desired weapon system and then an adjudicator would determine if the defensive weapon was effective in stopping the Harpy threat.

1. Defensive Weapon Listener Design

The ship in the DES model already incorporates a sensor to act as the ship's radar for detecting contacts. Therefore, the addition of a series of Listeners to act upon the information already being obtained by the ship's sensor would allow for the implementation of defensive weapons. Figure 12 demonstrates a potential Listener arrangement for a defensive weapon system implementation.

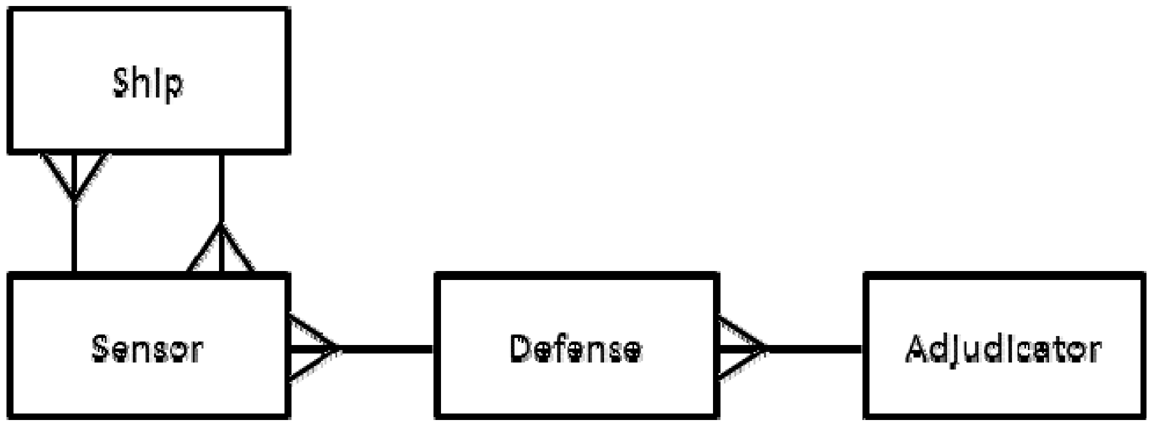


Figure 12. Potential Listener Implementation for a Defensive Weapon

The Listener arrangement of Figure 12 allows the detection of a contact by the ship's sensor to trigger the defenses of the ship. A class named Defense would then initiate to determine the nature of the contact, friend or foe, and launch defensive measures if needed. The Adjudicator Listener would then determine if the defensive weapon successfully eliminates the Harpy threat.

2. Defense Class Design

The Defense class associated with the Listeners of Figure 12 would consist of a Detection event from the ship's sensor which is then assessed by the Assess Threat event to determine if the detected contact (h) is a friend or a foe. If the contact is determined to be a foe then a defensive weapon (w) is launched to intercept the threat contact, see Figure 13.

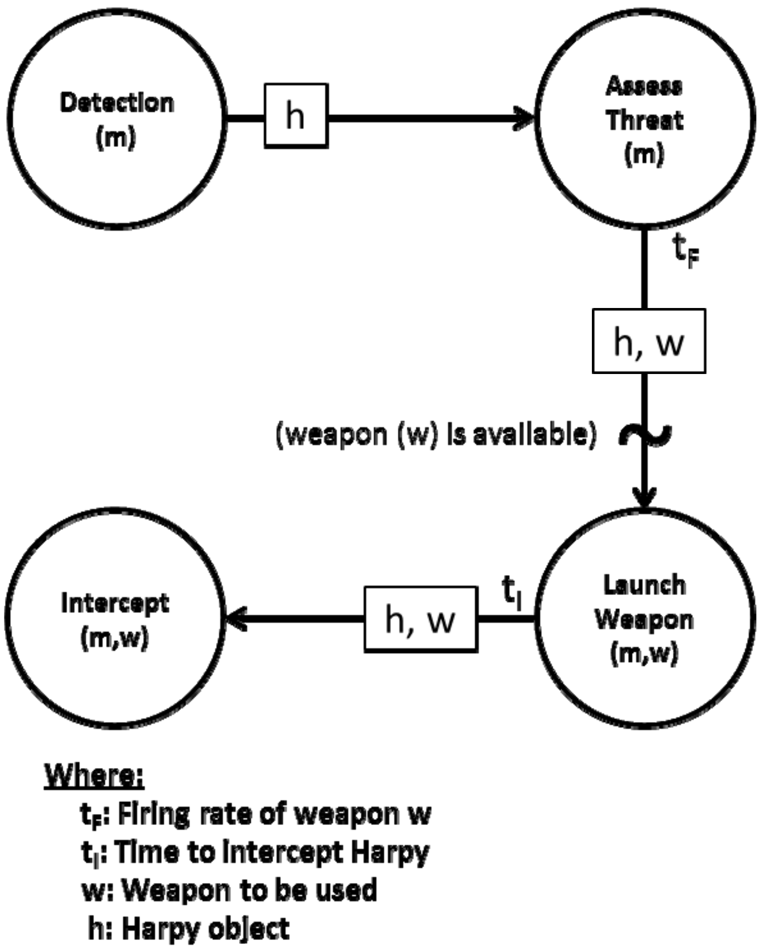


Figure 13. Potential Defensive Weapon Class Event Graph

Upon the defensive weapon (w) reaching the intercept with the contact, the Adjudicator of Figure 12 obtains a uniform random number and compares it to the probability of kill, P(kill), for the weapon. If the random number is less than P(kill) then the contact is destroyed, see Figure 14.

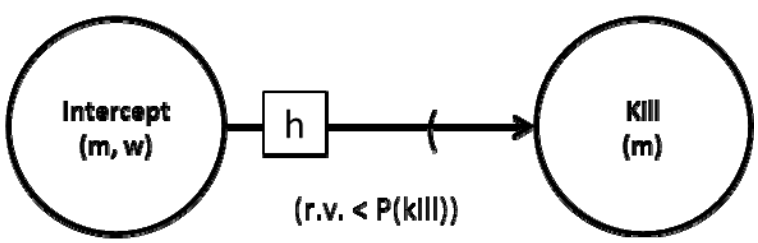


Figure 14. Potential Defensive Weapon Adjudicator Event Graph

The use of Listeners for the detection of a contact allows for the implementation of multiple defensive weapon systems simultaneously or based on a series of conditions, such as most effective weapon for a given range, or the number of munitions available for a specified weapon. The adaptability provided by using a defensive weapon class in this manner will also allow for weapons with different characteristics to function together, allowing for a much more varied test platform for testing various weapons and techniques for defeating a Harpy swarm attack.

V. DESIGN OF EXPERIMENT

As the goal of this research is to develop a baseline DES model to be used for future analysis of threats presented by the Harpy, as well as techniques to counter a Harpy swarm attack, the Design of Experiment (DOE) will be to determine the factors of a Harpy which are the most significant. This analysis is conducted to evaluate the base Harpy model's performance in simulating an attack and its ability to provide quantitative assessment of the attack against friendly defenses.

A. NEARLY ORTHOGONAL LATIN HYPERCUBE DESIGN

The seven parameters chosen to be varied for the DOE are the parameters associated with the actual operating characteristics of the Harpy as well as the speed at which the ship traverses the area. These parameters were chosen based on the expectation that they can provide the most insight for future research into defeating a Harpy swarm attack. Table 2 lists the DOE parameters obtained from the Nearly Orthogonal Latin Hypercube (NOLH) design, their associated minimum and maximum values based on unclassified comparable systems, and the determined 17 design points to be analyzed. Table 3 lists the parameters which will not be varied as well as their associated values.

low level	**10**	**2000**	**100**	**200**	**10**	**30**
high level	**20**	**9840**	**175**	**300**	**45**	**80**
decimals	**0**	**0**	**0**	**0**	**0**	**0**
factor name	**shipSpd**	**harpyCruiseAlt**	**harpyCruiseSpd**	**harpyDiveSpd**	**harpyBeamWidth**	**harpyPhit**
	13	9840	161	238	19	77
	11	3960	166	256	10	46
	11	5430	105	225	32	71
	12	6900	123	300	30	36
	18	9350	133	213	21	30
	20	4450	128	281	12	68
	16	3470	175	231	41	52
	16	8860	156	294	38	61
	15	5920	138	250	28	55
	17	2000	114	263	36	33
	19	7880	109	244	45	64
	19	6410	170	275	23	39
	18	4940	152	200	25	74
	13	2490	142	288	34	80
	10	7390	147	219	43	43
	14	8370	100	269	14	58
	14	2980	119	206	17	49

Table 2. Design Point Results (From Sanchez, 2005)

Parameter	Value
shipInitX	75 (nm)
shipInitY	0 (nm)
shipFinX	-75 (nm)
shipFinY	0 (nm)
patrolBoxXSize	25 (nm)
patrolBoxYSize	25 (nm)
patrolBoxDist	50 (nm)
harpyDetRange	50 (nm)
harpyDiveAngle	75 (degrees)
harpyQuant	54
replications	100

Table 3. DES Parameters Not Varied and their Associated Values

For more information about NOLH design and applications, refer to Sanchez (2005).

B. PERFORMANCE MEASURE AND SCENARIO REPLICATION

Each of the 17 design points of Table 2 was replicated 100 times for a total of 1700 data points. The response variable measured in each case was the number of Harpy UAVs that impacted the ship.

THIS PAGE INTENTIONALLY LEFT BLANK

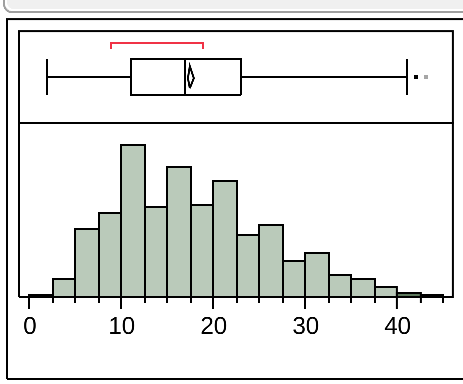
VI. ANALYSIS AND RESULTS

The analysis and results are based on unclassified parameters and therefore results using actual data may vary. The purpose of this analysis is to test the DES model and determine the factors with the most significance in order to gain understanding for areas of future research that can be pursued using this model.

A. BASIC STATISTICS

Figure 15 shows the basic statistics for the number of Harpy hits on the target. The results indicate that of the 54 Harpy UAVs present in each run of the simulation an average of 17.5 +/- 8 detected, intercepted, and successfully hit the ship.

Num Hits



Quantiles

100.0%	maximum	43
99.5%		39
97.5%		35
90.0%		29
75.0%	quartile	23
50.0%	median	17
25.0%	quartile	11
10.0%		8
2.5%		5
0.5%		3
0.0%	minimum	2

Moments

Mean	17.476471
Std Dev	7.9649046
Std Err Mean	0.1931773
Upper 95% Mean	17.855361
Lower 95% Mean	17.09758
N	1700

Figure 15. Distribution of Number of Hits

B. MAIN EFFECTS REGRESSION MODEL

A stepwise regression analysis of the six main effects was conducted generating a regression model. Figure 16 demonstrates a plot of actual hits by predicted plot of the regression model and Figure 17 shows the Summary of Fit (SoF) and Analysis of Variance (ANOVA) of the regression model using only main effects.

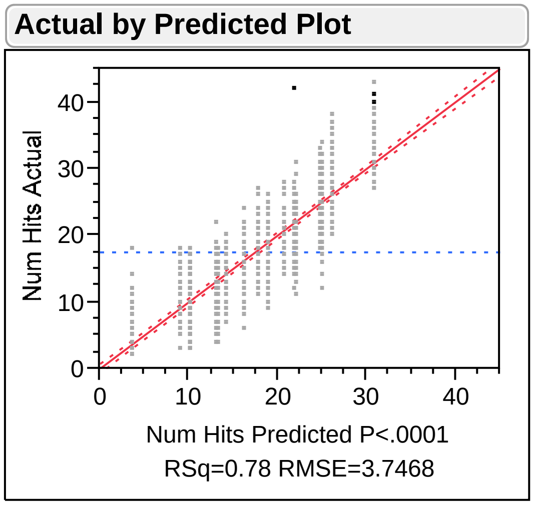


Figure 16. Actual by Predicted Plot of Main Effects Model for Number of Hits

Summary of Fit

RSquare	0.779226
RSquare Adj	0.778705
Root Mean Square Error	3.746848
Mean of Response	17.47647
Observations (or Sum Wgts)	1700

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	4	83988.17	20997.0	1495.636
Error	1695	23795.89	14.0	**Prob > F**
C. Total	1699	107784.06		<.0001*

Figure 17. SoF and ANOVA Results for Main Effects Model for Number of Hits

As Figures 16 and 17 both illustrate, the p-value of the main effect model is less than 0.0001, indicating that the model is statistically significant. The R^2 value of 0.78 indicates that 78% of the variability in the number of hits is accounted for by the model. The four statistically significant terms of the main effects model are shown in Figure 18 sorted by their importance to the response parameter, Number of Hits.

Sorted Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
P(hit)	32.210606	0.591498	54.46	<.0001*
Beam Width	0.4137338	0.008467	48.86	<.0001*
Ship Speed	-0.623631	0.029969	-20.81	<.0001*
Dive Speed	0.0369273	0.002969	12.44	<.0001*

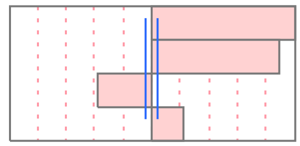


Figure 18. Sorted Parameter Estimates of Main Effects Model for Number of Hits

As seen from Figure 18, the most important factor when determining the number of hits that occur is the probability that a Harpy will actually hit the ship, P(hit). The second most important factor is Beam Width. The operational interpretation of this result is that the narrower the Beam Width is the less likely the Harpy is to detect the ship and therefore pursue it. Ship Speed also plays a role, the less time spent in the Harpy patrol area the fewer number of potential Harpy detections and therefore fewer potential Harpy UAVs intercepting the ship. These are validating results from an actual operational view point. The predicted regression equation for the main effects model is demonstrated in Equation 1.

$$\text{Number of hits} = -11.49 - 0.62 * (\text{Ship Speed}) + 0.04 * (\text{Dive Speed}) + 0.41 * (\text{Beam Width}) + 32.21 * P(hit) \quad (1)$$

Figure 19 is the distribution of residuals for the main effects regression demonstrating that the residuals are symmetrical and unimodal, both of which are desired characteristics.

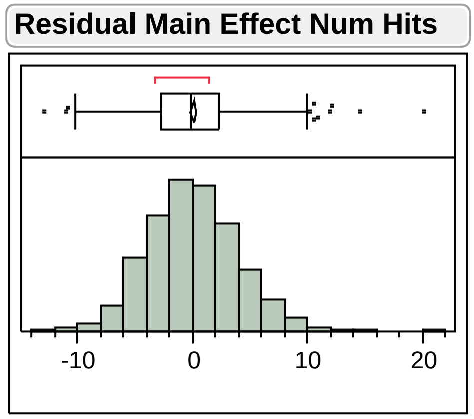


Figure 19. Residuals of Main Effects Model for Number of Hits

Due to the nature of the simulation just testing the baseline capabilities of the model the initial results of the main effect analysis all make intuitive sense. The intention of the model is not to gain an insight into employment tactics, rather to demonstrate that the model works and has the ability to provide a basis for future research.

C. FULL FACTORIAL REGRESSION

While the results of the main effects regression accounted for 78% of the variability the desire to account for more of the variability dictates that further regression analysis is required.

Using a full factorial design determines not only the main effects but also the interaction terms that have an influence on the model. Figure 20 demonstrates a plot of actual hits by predicted plot of the full factorial regression model and Figure 21 shows the Summary of Fit (SoF) and Analysis of Variance (ANOV) of the full factorial regression model.

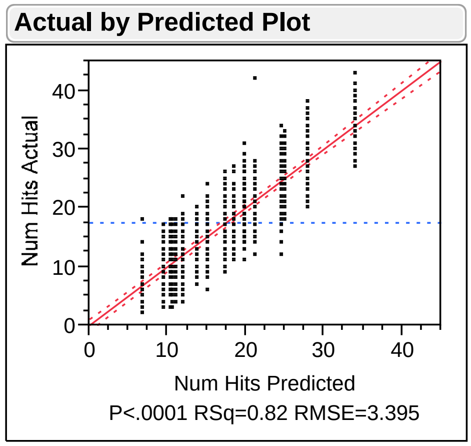


Figure 20. Actual by Predicted Plot for Full Factorial Regression Model for Number of Hits

Summary of Fit

RSquare	0.820025
RSquare Adj	0.818314
Root Mean Square Error	3.395014
Mean of Response	17.47647
Observations (or Sum Wgts)	1700

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	16	88385.60	5524.10	479.2680
Error	1683	19398.46	11.53	**Prob > F**
C. Total	1699	107784.06		<.0001*

Figure 21. SoF and ANOVA Results for Full Factorial Model for Number of Hits

Based on the results shown in Figures 20 and 21, the p-value of the full factorial model is less than 0.0001, indicating that the model is statistically significant. The R^2 value of 0.82 indicates that 82% of the variability in the number of hits is accounted for by the model. The 16 statistically significant terms of the full factorial model are shown in Figure 22 sorted by their importance to the response parameter, Number of Hits. Figure 23 is the distribution of residuals demonstrating that the residuals are symmetrical and unimodal, both of which are desired characteristics.

Sorted Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
(Beam Width-27.5294)*(P(hit)-0.55059)	1.7405854	0.176449	9.86	<.0001*
(Cruise Alt-5920)*(Beam Width-27.5294)	-5.573e-5	7.029e-6	-7.93	<.0001*
(Cruise Alt-5920)*(Dive Speed-250.118)*(Beam Width-27.5294)*(P(hit)-0.55059)	2.9122e-5	4.019e-6	7.25	<.0001*
(Cruise Alt-5920)*(P(hit)-0.55059)	-0.005545	0.000945	-5.86	<.0001*
P(hit)	74.681608	17.84786	4.18	<.0001*
Beam Width	3.2921086	1.213049	2.71	0.0067*
Dive Speed	1.0568509	0.430788	2.45	0.0143*
(Dive Speed-250.118)*(Beam Width-27.5294)*(P(hit)-0.55059)	-1.000042	0.420676	-2.38	0.0176*
(Cruise Alt-5920)*(Dive Speed-250.118)*(Beam Width-27.5294)	-0.00002	8.403e-6	-2.37	0.0177*
(Cruise Alt-5920)*(Beam Width-27.5294)*(P(hit)-0.55059)	-0.023035	0.009723	-2.37	0.0179*
(Cruise Alt-5920)*(Dive Speed-250.118)*(P(hit)-0.55059)	0.0050809	0.002147	2.37	0.0181*
Cruise Alt	-0.00953	0.004055	-2.35	0.0189*
Ship Speed	7.8091327	3.569743	2.19	0.0288*
(Dive Speed-250.118)*(P(hit)-0.55059)	-0.044143	0.023498	-1.88	0.0605
(Dive Speed-250.118)*(Beam Width-27.5294)	-0.001878	0.001661	-1.13	0.2584
(Cruise Alt-5920)*(Dive Speed-250.118)	3.6743e-6	7.04e-6	0.52	0.6018

Figure 22. Sorted Parameter Estimates for Full Factorial Regression of Number of Hits

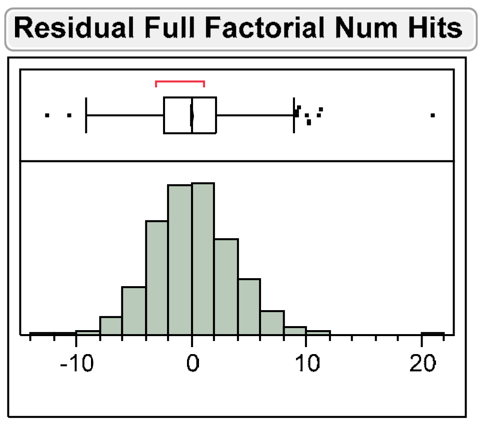


Figure 23. Residuals of Full Factorial Model for Number of Hits

The results of the full factorial model indicate that while the four parameters of significance from the main effects model still play a role the addition of the Harpy cruising altitude, harpyCruiseAlt, has been incorporated as well as interactions between the factors of significance. Figure 24 is the interaction plot associated with the full factorial regression. The interaction plot indicates that Dive Speed does not interact with any of the other parameters. The interaction between P(hit) and Beam Width can most likely be explained by the idea that as Beam Width increases the likelihood of detecting the ship also increases thereby increasing the chance of a Harpy attacking and subsequently hitting the ship, and as P(hit) increases the Harpies that have already detected the ship will have a higher probability of hitting the ship. The interaction between P(hit) and Cruise Alt does not have a readily apparent answer and further research would need to be conducted to determine the reason for the interaction.

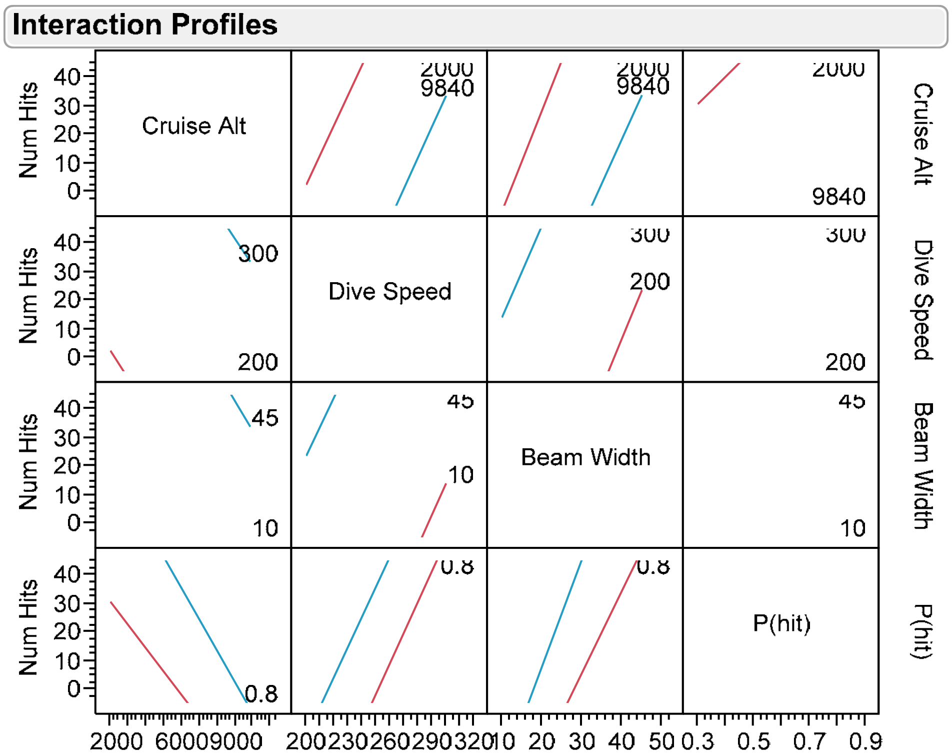


Figure 24. Interaction Profiler of Full Factorial Model for Number of Hits

D. PARTITION TREE MODEL

The use of a partition tree allows for the identification of key parameter levels which account for the specified variability in the response variable, Number of Hits. Figure 25 is the partition tree for the simulation results using seven splits resulting in an R^2 of 0.795, or 79.5% of the variability is accounted for. While further splits can give more insight the diminishing returns on R^2 indicate that seven splits is a good representation of the model. The partition tree indicates that the first key parameter level is a Beam Width of 32 degrees. The second and third splits are at a P(hit) of 0.61 and

P(hit) of 0.55, respectively. Further splits indicate that when Beam Width is less than 32 degrees then Dive Speed is important while when Beam Width is larger than 32 degrees the cruising altitude is important.

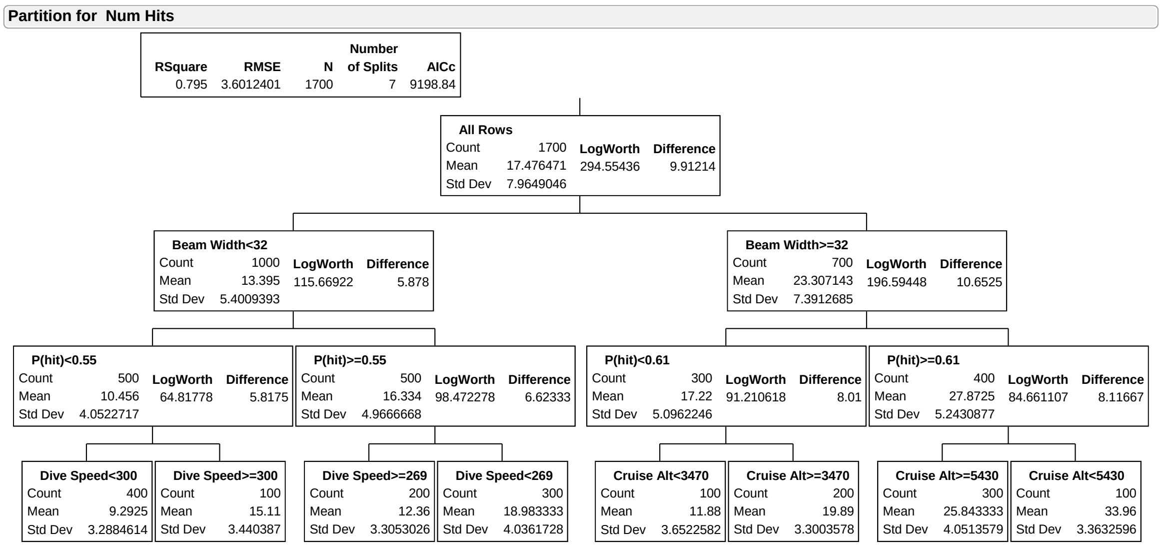


Figure 25. Partition Tree Model for Number of Hits

The results of the partition tree indicate that the two most important factors are the Beam Width and the P(hit), just as they were in the main effects model. However, unlike the main effects model, the partition tree indicates that the Cruising Altitude is also an important factor where as the main effects model indicates that Ship Speed is more important. Ship Speed does not have a role in the partition tree until split 12, which results in a R^2 of 0.819, an improvement of only 0.024 from the 7 split design of Figure 25. The interpretation of the partition tree is that the largest portion of the variability is accounted for by the Beam Width at a value of 32 degrees, followed by the P(hit), i.e., if a Harpy is able to detect the ship then the chance of the Harpy hitting the ship is at its highest. Operationally, this interpretation makes sense because without the ability to detect the ship a Harpy will continue to loiter until running out of fuel. Therefore, all other factors are contingent on the ability of a Harpy to detect the ship to initiate the attack.

VII. CONCLUSIONS AND FUTURE WORK

A. CONCLUSIONS

The goal of this research was to develop a DES model to continue the research performed by LT Kaiser, LT Hafer, and LT Taylor. NSS was unable to function as the simulation tool of choice for future research; therefore, a new one had to be developed. Through the use of Simkit a DES model was developed and implemented using the baseline scenario of a ship traveling through an area patrolled by Harpies. Unlike the rigidity of NSS, the Simkit model allows for adjustment and addition to all areas of the simulation. The use of Event Graphs to represent the simulation allows future researchers to identify where modifications and/or additions are necessary in order to achieve the desired outcome. The adaptability of the DES model allows for the testing of techniques as well as weapons system which may not currently exist, the addition of conditional logic can also test to see which combination of defensive weapons or techniques produce the best results when defending against a Harpy swarm attack.

While the results of the main effects model, the full factorial model, and the partition tree model help identify the parameters of most concern from the perspective of the ship the results may not be an accurate representation when actual CLASSIFIED values are used. All three models identify Beam Width and P(hit) as the most significant parameters of the simulation and as such these parameters identify a starting point for future research. The results are operationally validating as Beam Width increases probability of detection and therefore probability of hit given a detection. An increase in probability of hit given a detection logically results in an increase number of hits.

B. FUTURE WORK

Future research involving programmed patrol paths would require modification of the ArcCookieCutterMediator in order to better reflect the proper direction of the Harpy sensor. Java’s current implementation of mathematical arc functions is not adequate for the task, but other areas of research are still available for the simulation.

The DES developed for this research is a base attack model intended for use as a test platform for future research. Combating a Harpy swarm attack may take many forms; electronic emissions control (EMCON), various weapon systems, and cooperative engagement are just a few of the potential directions future research can proceed. As previously discussed, a basic assumption about the operation of the Harpy sensor relies on the movement being random.

Improvement in the functionality of the DES can take the form of adding components for defensive weapons systems, both current and future prototypes, and then determining the combination with the best results. Defensive weapons systems can also be added to determine if they are even a viable option or whether a different system is a better choice. Future research into the viability of countermeasures, and their optimal employment is another area where the DES has potential to develop new doctrine and techniques for combating a Harpy swarm attack. Currently, the DES model is designed around only one ship being in the area but little modification would be required to add any number of ships, presenting another avenue for research into cooperative tactics both for defending against a Harpy swarm as well as reducing their effectiveness at area denial.

In each of these cases, the use of this DES model can aid a researcher by being the test platform used. The previous research conducted by LT Kaiser, LT Hafer, and LT Taylor all contain suggestions for future research for combating a Harpy swarm attack which could be performed using this DES model.

LIST OF REFERENCES

Buss, A. (2001a). Basic event graph modeling. *Simulation News Europe, 31*, 1–6.

Buss, A. (2001b). Discrete event programming with Simkit. *Simulation News Europe, 32/33*, 15–25.

Hafer, William, J. (2010). *Analysis of the vulnerability of an Aegis destroyer against an Unmanned Aerial System (UAS) saturation attack in a multiple threat environment* (classified M.S. thesis). Naval Postgraduate School.

IHS Jane's: Defence & security intelligence & analysis. (2010, April 7). *IsraelAerospace Industries (IAI)/MBDA/Raytheon CUTLASS/Harpy/Harop/White Hawk.* Retrieved from http://search.janes.com.libproxy.nps.edu/Search/documentView.do?docId=/content1/janesdata/binder/jema/jaem0279.htm@current&pageSelected=allJanes&backPath=http://search.janes.com/Search&Prod_Name=JEMA&keyword=#toclink-j2121188555729765

Kaiser, Chad, S. (2008). *Analysis of the vulnerability of an Aegis destroyer to a kamikaze unmanned aerial systems saturation attack* (classified M.S. thesis). Naval Postgraduate School.

Sanchez, S. M. (2005). NOLHdesigns spreadsheet. Retrieved from: http://diana.cs.navy.mil/SeedLab

Taylor, Jeremy, B. (2010). *Analysis of the defensive capabilities of mutually supporting Aegis destroyers against a HARPY unmanned aerial systems swarm attack* (classified M.S. thesis). Naval Postgraduate School, Monterey, CA.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Arnold Buss
   Naval Postgraduate School
   Monterey, California

4. Jeffrey Kline
   Naval Postgraduate School
   Monterey, California

5. Brandon Cobb
   Naval Postgraduate School
   Monterey, California