8-2007

Study of Interaction Between Mexican Free-tailed Bats (Tadarida Brasiliensis) and Moths and Counting Moths in a Real Time Video

Haritha Kolli
University of Tennessee - Knoxville

To the Graduate Council:

I am submitting herewith a thesis written by Haritha Kolli entitled "Study of Interaction Between Mexican Free-tailed Bats (Tadarida Brasiliensis) and Moths and Counting Moths in a Real Time Video." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Engineering.

Hairong Qi, Thomas G. Hallam, Major Professor

We have read this thesis and recommend its acceptance:

Donald W. Bouldin

Accepted for the Council:
Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a thesis written by Haritha Kolli entitled "Study of Interaction Between Mexican Free-tailed Bats and Moths and Counting Moths in a Real Time Video." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Engineering.

Hairong Qi
Hairong Qi, Major Professor

Thomas G. Hallam
Thomas G.Hallam, Co-Major Professor

We have read this thesis
and recommend its acceptance:

Donald W. Bouldin

Accepted for the Council:
Carolyn R. Hodges
Vice Provost and Dean of the
Graduate School

(Original Signatures are on file with official student records.)

Study of Interaction between Mexican Free Tailed Bats (Tadarida Brasiliensis) and Moths and Counting Moths in a Real Time Video

A Thesis
Presented for the
Master of Science Degree
The University of Tennessee, Knoxville

Haritha Kolli
August 2007

Dedication

This Thesis is dedicated to:

My Parents

My Sister

And

My Brother-in-Law

Acknowledgement

First and foremost, I would like to sincerely thank Dr. Thomas G.Hallam for giving me this opportunity to work on this NSF project and supporting me financially through out my Masters program. His patience, encouragement and valuable guidance made me learn a lot while working on my thesis. I would like to thank Dr. Hairong Qi for introducing me to the world of image processing and guiding me through out my Masters. I would also like to thank Dr. Donald W.Bouldin for reviewing my thesis.

Special thanks to Dr. Dobri Dimitrov and Jeff Nichols for sharing their knowledge and help me in debugging my code. I also thank Dr.Paula Federico for her valuable suggestions and discussions. Special thanks to Kim Kennard for bringing the moth counting problem to me.

I would like to thank my friends Suman, Prem, Raghul, Kiran and Lakshmi for their encouragement, support, care and for helping me get through the difficult times.

Last but not the least I thank my parents, sister and brother-in-law, whose love and understanding are always essential to my success. My family has been my source of inspiration.

Abstract

Brazilian free tailed bats (*Tadarida brasiliensis*) are among the most abundant and widely distributed species in the southwestern United States in the summer. Because of their high metabolic needs and diverse diets, bats can impact the communities in which they live in a variety of important ways. The role of bats in pollination, seed dispersal and insect control has been proven to be extremely significant. Due to human ignorance, habitat destruction, fear and low reproductive rates of bats, there is a decline in bat populations. *T.brasiliensis* eats large quantities of insects but is not always successful in prey capture. In the face of unfavorable foraging condition bats reduce energy expenditure by roosting. By studying the interaction between bats and adults insects along with the associated energetics, we estimate the pest control provided by bats in agro-ecosystems to help understand their ecological importance. To visualize the interaction between bats and adult insects, a simulator has been designed. This simulator is based upon an individual based modeling approach. Using the simulator, we investigated the effect of insect densities and their escape response on the foraging pattern of bats.

Traditionally synthetic pesticides were used to control pest population. But recently the use of transgenic crops has become widespread because of the benefits such as fewer pesticide applications and increased yield for growers. To study the effect of these transgenic crops on moth densities and subsequently on bats foraging activity, videos were recorded in the fields at Texas. To count the moths in the videos, we utilized image segmentation techniques such as thresholding and connected component labeling. Accuracy up to 90% has been achieved using these techniques.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

1.1 Bat Ecology

Worldwide there are more than 90 species of insectivorous free tailed bats. Most occur in tropical and subtropical climates. Six species occur in the United States –one in Southeast, the remainder in the southwest, mostly in arid regions. Free tailed bats are easily recognized because at least a third of their tail protrudes beyond the membrane that connects the legs and tail. Brazilian free tailed bats (*Tadarida brasiliensis*) are among the most abundant and widely distributed in the southwestern United States in the summer (Lee and McCracken 2005). It has been estimated that over 100 million bats migrate into south central Texas where they roost from April to October (Lee and McCracken 2001,Wahl 1993).

Bats are voracious predators of night flying insects and many of the insects are eaten by bats in abundance. Noctuid moths (Thompson 1982; Robinson 1990), are major agricultural pests known to engage in seasonal, long distance migrations. The prey of *T.brasiliensis* includes adults of several Lepidopteron species in the family Noctuidae (Lee and McCracken 2002,Lee and McCracken 2005), whose larvae are known agricultural pests, such as fall armyworm (*Spodoptera frugiperda*), cabbage looper (*Trichoplusia ni*), tobacco budworm (*Heliothis Virescens*) and corn earworm or cotton bollworm (*Helicoverpa zea*). Because of their high metabolic needs and diverse diets, bats can impact the communities in which they live in a variety of important ways. The role of bats in pollination, seed dispersal and insect control has been proven to be extremely significant (Council 1999). In addition, bat guano (feces) is often used as a fertilizer. The bollworm (*H.Zea*) causes over $1.5 billion in annual losses in the United States from crop damage and pest control (King and Rogers 1986). The economic value of the pest control service provided by *T.brasiliensis* in south central Texas in a recent study is estimated to be up to $741,000 per year (Cleveland, Betke et al. 2006). Unfortunately, human ignorance, fear, myth, habitat destruction and bats' low reproductive rates continue to contribute to the decline of many bat species worldwide. Ecologically the extirpation of bats from an area can leave plant communities that rely on bats for pollination and seed dispersal without reproductive capabilities, as well as increase the need for use of chemical pesticides – potentially threatening entire ecosystems (Andrewartha and Birch 1954).

1.2 Motivation

Large numbers of insectivorous Brazilian free tailed bats reside in the south and south-central United States from spring to early fall. The sizes of their colonies range from several thousands in many man made structures to tens of millions in some limestone caves (Davis, Herreid et al. 1962). For years, scientists have been tracking the migration and arrival of corn earworms and other pests into south Texas from Mexico (Wolf, Westbrook et al. 1990). The timing of these events closely correlates with the flight patterns, colony locations and foraging range of Brazilian free-tailed bats (Lee and McCracken 2001,Lee and McCracken 2002).Lee and McCracken (2005) analyzed bat fecal material and found that the temporal and seasonal variation in moth consumption in the diet of Mexican free tailed bats is strongly correlated with the availability of migratory moth populations, particularly corn earworm moths. This co-migration has led to studies of the movement of the bats from the emergence, their nature of flight, the movements of each bat inside an emergence column and then finally the amount of insects consumed by the bats.

1.3 Thesis goals

Individual–based modeling is a reductionist technique for describing ecological systems. Individual models are bottom up approaches that start at the bottom level of population ecology, that is, at the individual level (Deangelis and Gross 1992). Individual based models have the potential to determine what individual properties and what elements of an individual's performance are essential for generating the characteristic features of the overall population dynamics. This approach includes the possibility of taking spatial dynamics and the dynamics of abiotic factors explicitly into account. To study the interaction between adult insects and bats in this thesis, an individual based modeling approach has been used.

1.4 Thesis Outline

Chapter 2 reviews the social biology of bats and the rules proposed by Aruna Raghavan (Raghavan 2005). The rules proposed to visualize the foraging behavior of bats and insect behavior has been discussed. In Chapter 3, the rules are discussed in detail. The mathematical models of each rule are delineated in detail. The pseudocode for each rule is presented. The movements of the bats and insects according these rules are visualized using the simulator designed for this purpose. Snapshots of the interface are used to describe the features of the simulator. In Chapter 4, we investigated the effects of variable insect densities and insect's ability to escape on the foraging behavior of bats. The results obtained using the simulator have been tabulated. In Chapter 5, an algorithm developed to automate counting of moths in a video file has been discussed in detail. The algorithm has been implemented on several video files of different lengths to evaluate the accuracy of the algorithm.

CHAPTER 2

Background

Bats are the only flying mammals. Their flight has aroused the interest of many researchers. A typical flight pattern observed during an emergence from a cave or bridge is column formation. After moving in a column for considerable amount of time and before starting to forage, bats separate into sub flocks. The rules that mimic bat flight are explained later in this chapter.

The most widespread resources of free tailed bats are insects and other small arthropods. Vision is of limited use for tracking small, mobile, aerial prey in the dark or unpredictable lighting levels. In contrast, echolocation is effective for this purpose, but bats need to produce intense ultrasonic pulses in order to receive audible echoes from targets as small as insects. Due to high energy demands bats consume large quantities of insects. Bats are not always successful in prey capture.

There are several factors that influence the foraging behavior of bats. It is not a feasible strategy for bats to continue foraging throughout the night when they have not captured a considerable amount of insects, as energy involved in flight is not negligible. Anthony, Stack et al. (1981) observed that bats reduce the energy expenditure by roosting in the face of unfavorable foraging conditions. The factors influencing the foraging behavior of bats are explained in this chapter.

The noctuid insect pests that are economic detriments to agriculture are generally strong fliers that can maintain flight speeds of 2-6 km\hr in still air (Beerwinkle, Lopez et al. 1994). They can move several kilometers in one night without wind assistance; this contributes greatly to enhanced pest status (Beerwinkle, Lopez et al. 1995). Beerwinkle, Lopez et al. (1995) observed that insect densities was high near ground level, the densities decreased non-linearly with increasing altitude and they considerably reduced at altitudes above 800m at College Station, TX. The mathematical rules that mimic insect behavior are explained later in this chapter.

2.1 Energetics of Bats

Energy demands are high for bats due to small body size, high basal metabolic rates and costs of flight, demand increases dramatically for females during reproduction. The females during lactation, may ingest up to two thirds of their body mass each night (Kunz, Whitaker et al. 1995). Nocturnal foraging flights of Brazilian free-tailed bats cover no more than an 80 km radius from their cave habitats (Davis, Herreid et al. 1962). With these flight capabilities for migration, homing ability and foraging, the Brazilian free tailed bat can fly significant distances at high altitudes to visit foraging areas each night.

Lee and McCracken (2002) and Lee and McCracken (2005) performed fecal analysis of bats bat guano and found that about 31% of the Brazilian free tailed bat's diet consists of the order Lepidoptera . Wolf, Westbrook et al. (1990) documented high levels of foraging activity and consumption of insects by bats at altitudes of 200- 1200m where the bollworm population density is high. Given the extremely heavy densities of *H. Zea*, and the high energetic needs of Brazilian free tailed bats, a strong correlation could be expected between their ecological and economic services and their relative abundances over the landscape.

2.2 Echolocation

Echolocation is a complex highly evolved process that has given bats the ability to exploit an ecological niche closed to many animal groups- the night sky. Though echolocation is not unique to bats, it has reached its evolutionary peak in these mammals. The bat builds a sound picture of its immediate environment analyzing echoes of its own emitted sound waves. Echolocation is used by bats for detecting, tracking, and evaluating air borne prey, feeding almost exclusively on flying insects (Kunz 1982). Echolocation calls of bats vary in design and echolocation often reflects the sensory attributes of bats (Fenton 1990). Calls can be modified by individual bats according to conditions. For instance, in confined spaces, calls may become shorter and of broader bandwidth (Kalko and Schnitzler 1993). When searching for prey, bats emit search phase calls. On detection of the prey, pulse repetition rate increases and both pulse duration and interpulse interval decrease during the approach phase. They reach extreme values during the terminal phase of the buzz immediately prior to capture. During the search phase the echolocation calls emitted are designed for detecting targets, where as calls emitted during the approach phases are modified to provide more information on target location and type. During terminal phase, the function of signals emitted is to provide information on a prey's position immediately prior to capture. For the long-range detection of relatively large insects Brazilian Free tailed bats employ calls of low frequency (mostly <30 KHz) (Neuweiler 1990).

The echolocating bats can be detected using a system called Anabat II Bat Detector in the field. Employing the frequency division technique to make the bat calls audible, AnaBat II detects the ultrasonic calls emitted by bats. Using the internal loudspeaker or headphones the frequency divided signals can be heard, recorded, and analysed to assist in the identification and detection of the bat species. It is possible to permanently record the signals of bats onto a compact flash memory card for analysis later in the laboratory by using it together with the Anabat CF Storage ZCAIM (Zero-Crossings Analysis Interface Module).

For active monitoring applications, real time sonograms can be produced by connecting the ZCAIM output to a PC through its serial port. Alternatively, for simple detection and monitoring of bats the bat detector can be used entirely on its own. Fig 2.1 is an Anabat II Bat Detector.

2.3 Roosting

Bats occupy a wide variety of roosts in both natural and manmade structures. Roosting habits of bats are influenced by the diversity and abundance of roosts, the distribution and abundance of food, and an energy economy influenced by body size and the physical environment, especially temperature and humidity. It is a common habit of temperate zone insectivorous bats to occupy night roosts between foraging flights

Figure 2.1: Anabat II bat detector
(Image courtesy: www.titley.com.au)

(Anthony, Stack et al. 1981). In bats, due to high costs of flight, the metabolism expense of foraging is great (Thomas 1975). Therefore, a bat must have substantial amount of energy to survive even after performing tasks like feeding. Bats retreat to night roosts, when prey availability precludes high capture rates (Anthony, Stack et al. 1981). Although returning to roosts eliminates energy input for the duration of the roosting period, energy expenditure of clustered bats in these confined spaces is low. Thus bats minimize energy output by roosting.

2.5 Bat Behavior

A characteristic of Brazilian free tailed bats in Texas and Mexico is that they live together as a big colony inside caves. They often emerge from the cave in columns. The study of bat behavior is vital to determine whether there is a correlation between the bat flight and the foraging behavior of the bats. According to the emergence flight model proposed by (Raghavan 2005, Hallam et al. 2006), the primary rules followed by the bats are collision avoidance and individual predator avoidance. There are also other secondary rules such as community predator avoidance, sub flocking, flock forming and velocity matching. The primary and secondary rules model bat movement patterns. But the primary rules have higher priority than secondary rules. These rules focus on the movements of each bat to form the emergence column and dynamics within the column. The function of each rule implemented in the emergence flight model is:

Collision Avoidance Rule: According to this rule a bat tries to maintain a small predefined distance between its nearest neighbors in a prescribed neighborhood.

Velocity Matching Rule: In this rule, a bat tries to match the velocity of the bats in a small neighborhood around it.

Flock Forming Rule: In this rule, all the bats are restricted to movement within or on the boundary of the column.

Community Predator Avoidance Rule: The community predator avoidance rule is active only at the boundary of the column. The main aim of this rule is to move the bats inside the column as a group to avoid threats due to predators.

Individual Predator Avoidance Rule: The community predator avoidance rule made the groups of flyers in the boundary to be aware of their neighbors and save themselves as an individual from predators. The individual predator avoidance rule forces a bat to move towards the center of mass of the whole column.

Sub-Flocking Behavior Rule: The bats remain inside a column for an undetermined distance. As soon as the bats perceive they have out flown the reaches of the predators, they initially sub-flock.

To understand the foraging patterns of bats, two new rules - a Pursuit Rule and a Bat Satiation Rule are proposed.

Pursuit Rule: This rule mimics the foraging behavior of bats in the field. A cone of detection is defined. Insects which are within the cone of detection are detected and with a certain probability are captured and eaten by bats.

Bat Satiation Rule: The energy balance of the bat is calculated taking into account the energy spent in flight, in maintaining metabolism, in roosting and energy gained by consuming insects. Depending on foraging success and energy balance, the bats tend to return to the roosts or cave. This rule illustrates when the bat decides to return taking into account the energy obtained from the insects eaten, the time elapsed since last insect eaten and flight time.

In the next chapter, we will discuss how these rules are implemented in the simulator.

2.6 Insect Behavior

The study of insect behavior is vital in understanding the foraging pattern of the Brazilian free tail bats and to facilitate the development of improved regional area-wide management and control strategies. Taking advantage of wind flow many insect species have the ability to fly to new habitats that are more conducive to successful reproduction and survival. The rules implemented to model the movement of adult insects in the field are – collision avoidance, insect landing and insect migrating.

Collision Avoidance Rule: According to this rule, the adult insects maintain a predefined distance between adult insects in the neighborhood. The distance of separation is given by the user. If the distance of separation between the current insect and insects in neighborhood is less than the given distance of separation, the probability of collision is high. To avoid collision the insects are moved away from each other by a predetermined distance.

Insect Migrating: Migration can be regarded as an adaptation to escape predation, to reduce competition, to exploit periods of resource abundance, to avoid winter cold, or to leave excessively dry and wet seasons (M A Rankin and J C A Burchsted 1992). The insect migrating rule simulates the migrating behavior of insect. In this rule the insects move above an altitude of 800 units in the positive y axis of the 3D model used.

Insect Foraging: Insects move towards and across the agricultural crop to find food sources. This rule simulates the movement of insects within the field. In this rule, the insect's motion is limited to a height of 400 units on the positive y axis.

In the next chapter we will discuss how the rules are implemented in the simulator.

Chapter 3

Simulation

A complex image of a population of objects can be generated by modeling the simple behavior of each individual object and the interaction between objects. This approach is termed by Craig W. Reynolds as behavioral animation. Reynolds noticed that scripting the paths of a large number of individual objects such as flock of birds is a very difficult and tedious task. He demonstrated that behavioral animation is a more efficient and robust way to accomplish this task. The basic idea of behavioral animation is that the complex paths can be generated by simulating these models.

The main focus of this chapter is to explain the mathematical implementation of the bat and insect movement rules that are implemented in the models. The concept behind the simulation is to design a computational model of the interaction between bats and insects. The flight pattern of the insects has been also simulated separately in the INSECTOIDS module. There is a separate function in the program that makes the bats and insects move constantly along the desired direction and the rules guide the bats and insects in taking the direction that it may reflect in real life.

3.1 Explanation of Data Structure

Separate classes have been defined for the bats and the insects. The variables associated with the bats and insects are declared in each class. The variables and functions in each class control the movement of the bats and insects.

BoidsFlyer is the class defined for the bats. The variables previously defined are - location, speed and heading. Timing details of flight, energy, insects eaten and label of the bat are new variables introduced to study bat–insect interactions.

The location variables are x, y, z which are the coordinate values of the bat along the three dimensional axes. The heading variables are hHeading and vHeading, which gives the horizontal and vertical headings of each bat. The timing details of the model flight variable consists of flightStart, flightEnd, firstInsectEaten and lastEatenTime of the bat. These variables enable estimation of the time spent by the bat in seconds. The flightStart variable stores the time at which the bat started flight. The flightEnd variable stores the time at which the bat ends flight. The firstInsectEaten variable stores the time at which first insect is eaten. The lastEatenTime variable stores the time at which last insect was eaten. The insectEaten variable tracks the number of insects eaten by the bat. The energy variable is denoted by batsEnergy and stores the energy of each bat at each time iteration. The energetics of the bat is determined by the dynamic energy budget, which are the gains minus losses. The energetic losses of the bat include energy used by commuting to the field, foraging in the field, roosting, maintaining its metabolism while gains of energy occur by eating insects. The label variable stores a unique number for each bat used for identification purposes. The variables defined above are modified by the different interface functions defined by the boidsFlyer class, which reflect the movement of the bats. The functions are:
The pursuitEat Function: This function models how the bat pursues an insect. With a certain probability, the bat captures and feeds on the insects

present within its cone of detection. The cone of detection is the region within which bats can detect moths and probability of feeding on them is high.

The batSaturation Function: This function is used to calculate when the bat has optimum energy to return to the cave to feed its pup. The decision is based on time spent by the bat in the field, the time elapsed since last insect eaten and the current number of insects eaten.

InsectFlyer is the class defined to mimic the insects' flights. The variables defined are – location, speed, heading and label. The location variables are x, y, z which are the coordinates values of the insect along the three dimensions. The heading variables are hHeading and vHeading, which gives the horizontal and vertical headings of each insect. The speed variable stores the speed of the insect. The label variable stores a unique number for each insect used for identification purpose. The variables defined above are modified by the different interface functions defined by the InsectFlyer class which reflect the movement of the insects. The functions are:

The collisionAvoidance Function: This function allows an insect to avoid the collision with another insect by first calculating the distance of separation between the insect and its neighborhood insects. If the distance of separation is less than a specified value, the insects are moved apart to avoid collision.

The insectsMigrating function: In this function the insect's motion is limited to altitude of 800 units to 1200 units in the 3Dimensional space.

The insectLanding function: In this function the insect's motion is limited to an altitude less than 400 units in the 3 Dimensional space. Collision with trees and the landscape are avoided.

Next we explain the auxiliary functions which help in the motion of bats and insects according to the above described functions:

The moveAway Function: If the distance of separation between the bats or insects is less than collision distance this function moves the position of the bat/insect from the nearest neighbor. It is called from collisionAvoidance Function.

The moveObject Function: This function updates the position of the bat/insect as long as the simulation is running.

The pseudo code for each function will be given in the next section.

3.2 Mathematical Interpretation

3.2.1 Pursuit Rule

Brazilian free tailed bats use echolocation for orientation and to capture prey. Echolocation is a comprehensive mode of perception used for detecting, locating, and recognizing objects in the environment. In general, bat echolocation sounds consist of constant frequency (CF) and frequency modulated (FM) elements alone or in a combination of the two components.

In searching flight, Brazilian free tailed bats constantly monitor their location relative to insects. They use FM signal for this task. When flying near the obstacle they use broader bandwidth FM pulses to accurately localize and characterize the insect (Fenton, Racey et al. 1987). Depending on its efficiency the bat captures the insect. It starts searching for the next prey in the region around the last captured insect for maintaining a good capture rate. Figure 3.1 shows echolocation in bats. The pseudocode is given in the Figure 3.2.

3.2.2 Bat Satiation Rule

Roosts are used as places to rest between foraging bouts, to promote digestion and energy conservation, to provide retreats from predators and to serve as places that promote social interactions (Kunz 1982). In this rule, depending on the time spent in the flight, the time elapsed since last eaten insect and the number of insects eaten, the decision is made when the bat should return to the roost. In the function used for taking the decision, more weight is given to time elapsed since the last eaten insect. If the bat hasn't eaten an insect for a prescribed time, it returns to the roost. In my model, the prescribed time is 15 seconds. This is because there is a possibility of not eating insects in the future and the return prevents wasting more energy in foraging. The bat requires sufficient energy to return to its roost and lactating females need to feed their pups. The pseudo code is given in Figure 3.3

Figures 3.4, 3.5, 3.6, 3.7 are series of images to demonstrate the pursuit and bat satiation rule. In Figure 3.4 bats are introduced into the field having an insect density of 200. The three dimensional space visualized cannot be specified in standard units of length because DirectX doesn't provide units of conversion. Initial energy reserve is 200 KJ for each bat. In Figure 3.5 bats are successful in eating insects. In Figures 3.6 and 3.7 bats reach satiation level and, according to the rule, return to the cave. The number of insects eaten and energy of the bat can be determined from the slide bars present on the right hand side.

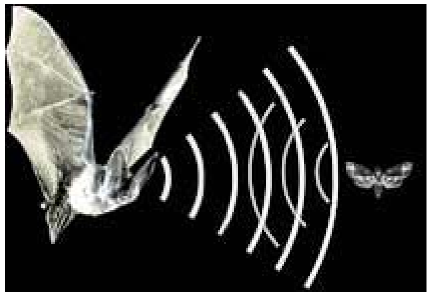

Figure 3.1: Pursuit Rule.
(Image Courtesy: http://www.tigerhomes.org/animal/images/bat-echolocation.jpg)

Data: b_i (an individual bat), i_i (an individual insect), d (distance of detection), α (angle of elevation)

Result: the updated position and heading of b_i and removal of i_i

For each bat b_i

For each insect i_i

    Given d and α calculate the cone of detection for the bat b_i

    Determine whether the insect i_i is within the cone of detection of bat b_i

    IF within cone of detection THEN

        Position of **bat** b_i = Position of insect i_i

        Vertical Heading of **bat** b_i = Vertical Heading of insect i_i

        Horizontal Heading of **bat** b_i = Horizontal Heading of insect i_i

        insect i_i removed

    END IF

END FOR

END FOR

Figure 3.2: Pseudo code for pursuit rule.

```
Data: b_i (individual bat)
Result: Return to roost or to continue foraging


FOR each bat b_i
        Calculate the value of the Satiation Function
        IF value greaten than 0.6 THEN
                bat b_i  returns to Roost
        ELSE
                   bat b_i  continues to Forage
        END IF
END FOR
```

Figure 3.3: Pseudo code for bat satiation rule.

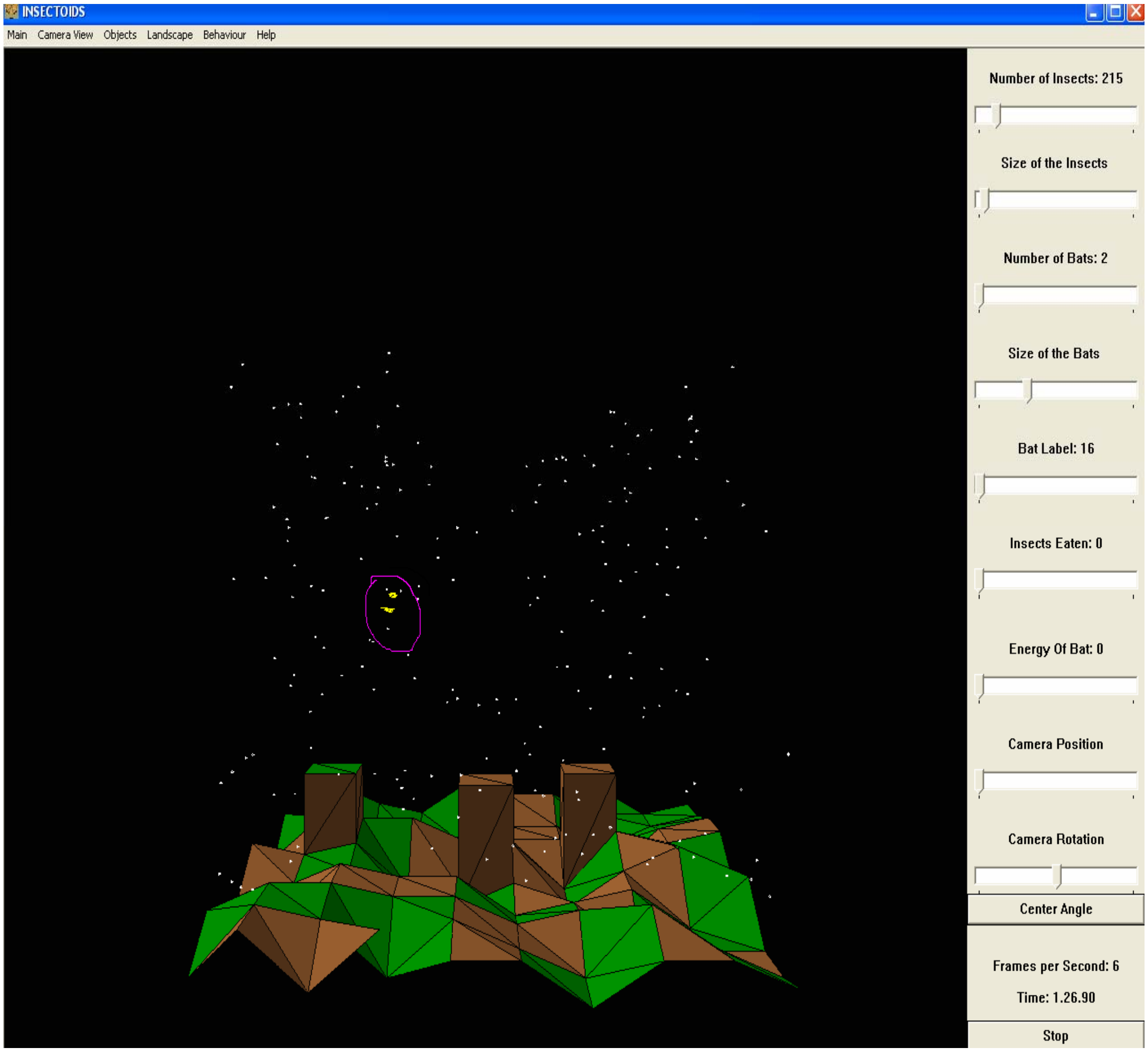

Figure 3.4: Bats are introduced in the field.

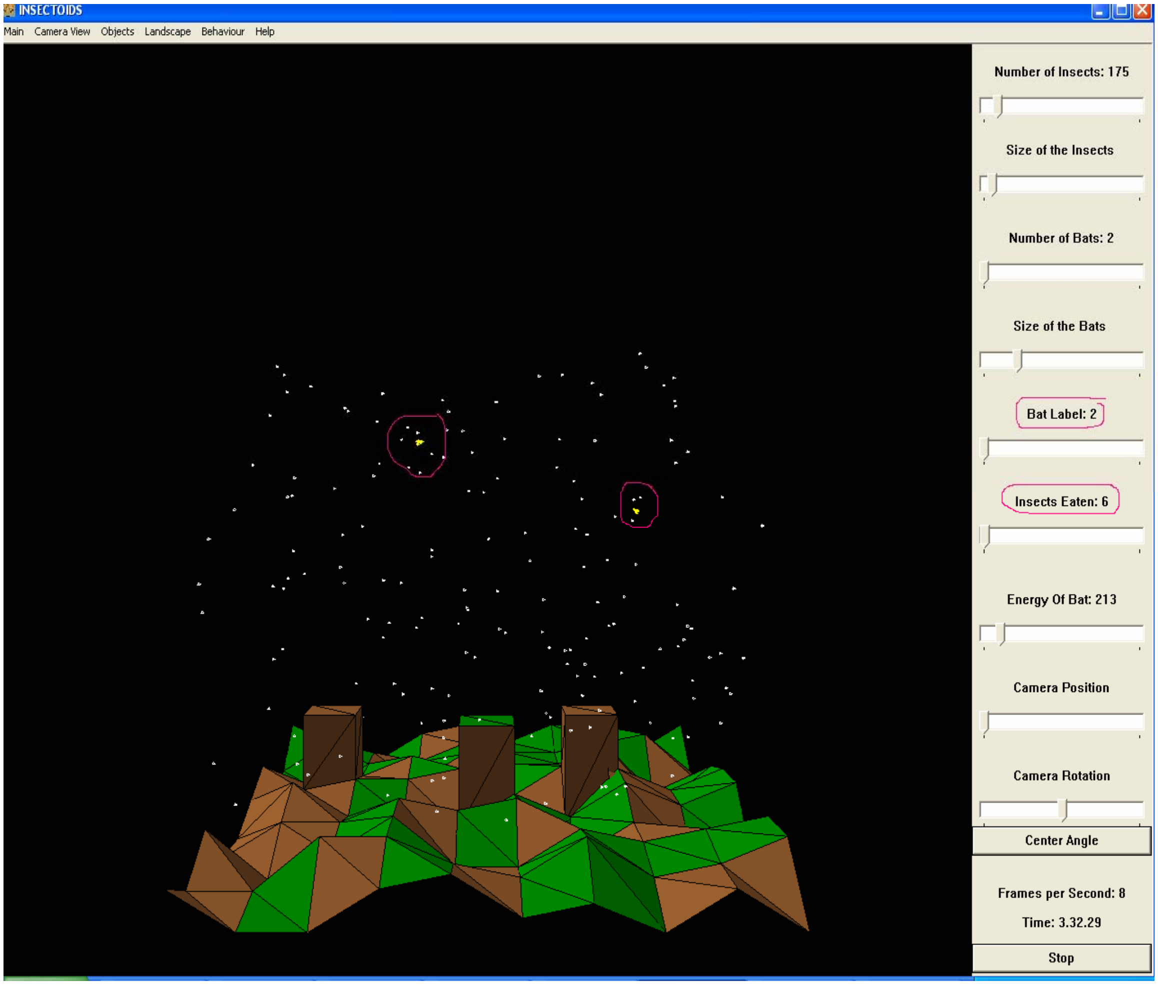

Figure 3.5: Bats are successful catching insects.

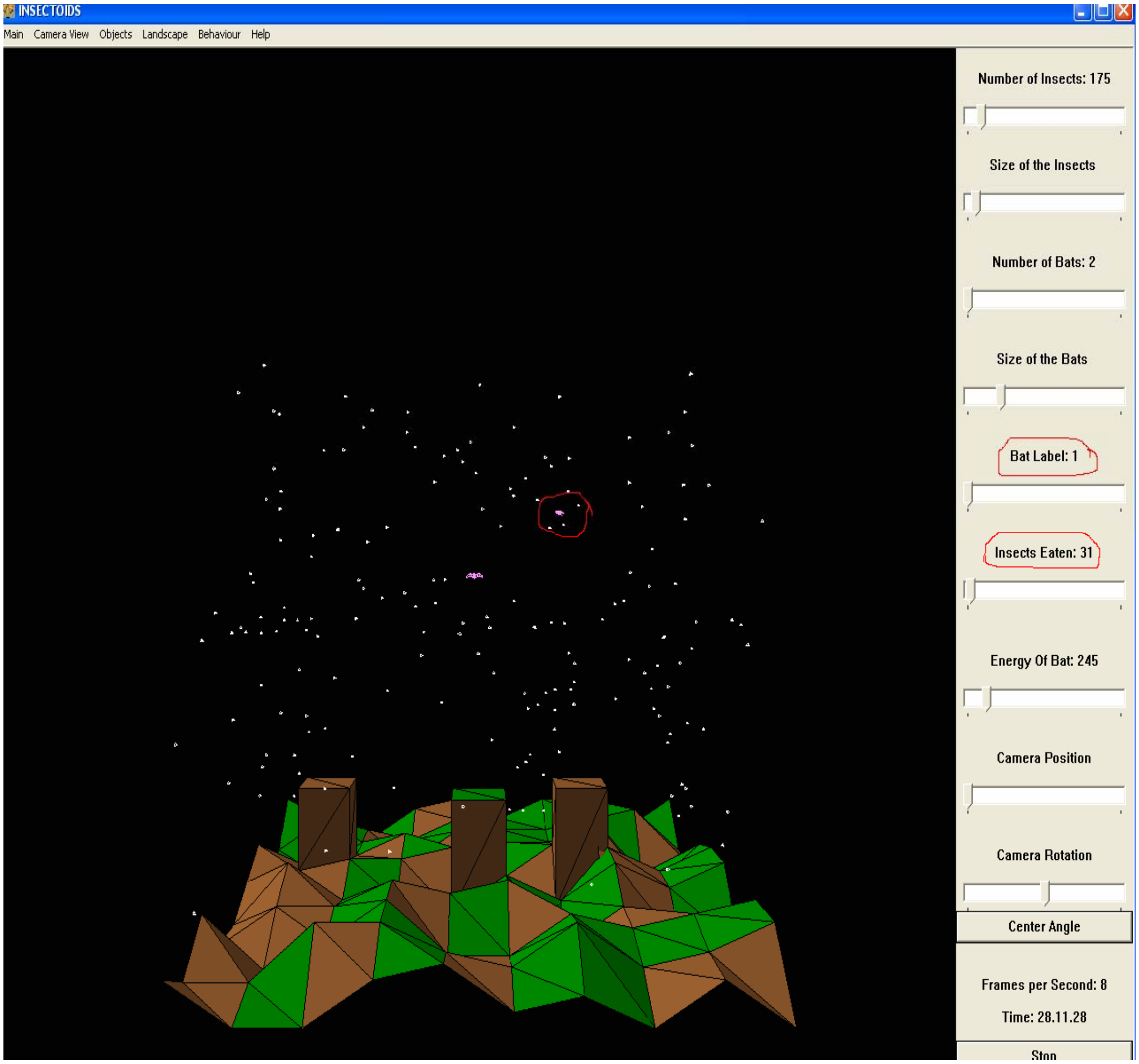

Figure 3.6: Bat 1 has reached satiation level.

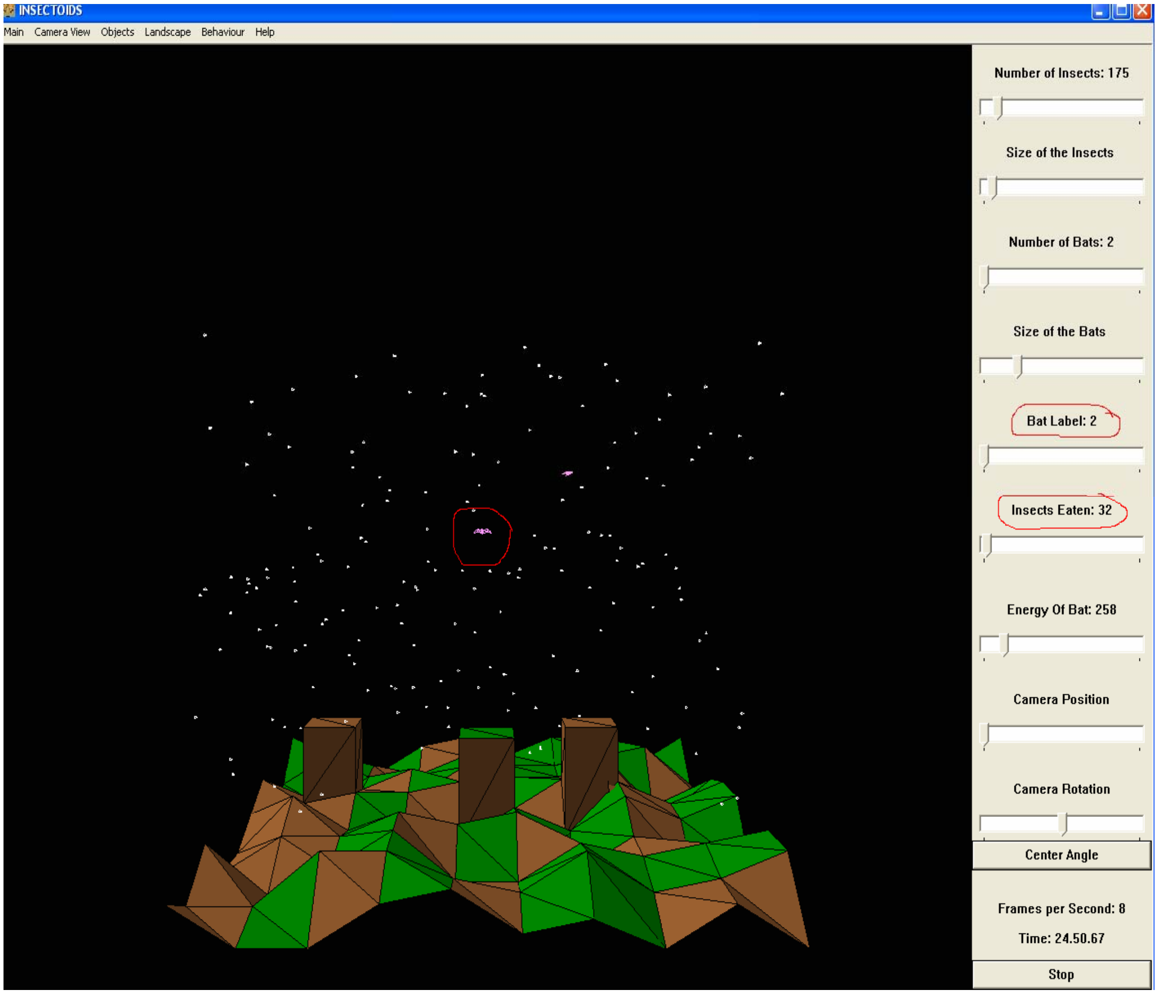

Figure 3.7: Bat 2 has reached satiation level.

3.2.3 Collision Avoidance Rule

This rule is implemented to avoid collision between insects. The distance of separation between an insect and its neighborhood insects is calculated. If the distance of separation is less than the given distance of separation then the insects are moved apart. The pseudo code for this rule is given in Figure 3.8. Figure 3.10 is a snapshot of this rule executed in the simulator.

3.2.4 Insects Migrating Rule

According to this rule, the insects which are below an altitude of 800 units in the 3Dimensional space used are forced to move up. This rule simulates the migration movement of the insects. The pseudo code is in Figure 3.9. Figure 3.11 is a snapshot of this rule implemented in the simulator.

3.2.5: Insects Foraging Rule

In this rule, the altitude of insects is limited to height of 400 units in the 3Dimensional space used. The height of every insect is checked. If it is greater than 400 units then it is updated to a random value below 400. This rule simulates the motion of the insects in the field feeding on the crops. The pseudo code for this rule is in the Figure 3.12. Figure 3.13 is a snap shot of the insects landing rule. In the enclosed region, the insects have landed in the agricultural field.

```
Data: i_i (an individual insect), i_ij (the neighboring insect), d (distance
of separation pre-defined)
Result: updated position of  i_i  after performing collision avoidance
rule.
Given each insect i_i
For each insect i_ij (i not equal to j)
        Calculate distance of separation between the insect
    i_i  and insect i_ij
        IF distance is less than d THEN
          Change the position of insect i_i
```

Figure 3.8: Pseudo code of collision avoidance rule.

```
Data: i_i (an individual insect)
Result: updated position of the insect i_i after applying the insects
migrating rule.

FOR each insect i_i
        IF y co-ordinate of i_i  less than 800
         y co-ordinate of i_i  is assigned a random value greater
    than 800
        END IF
END FOR
```

Figure 3.9: Pseudo code for insects migrating rule.

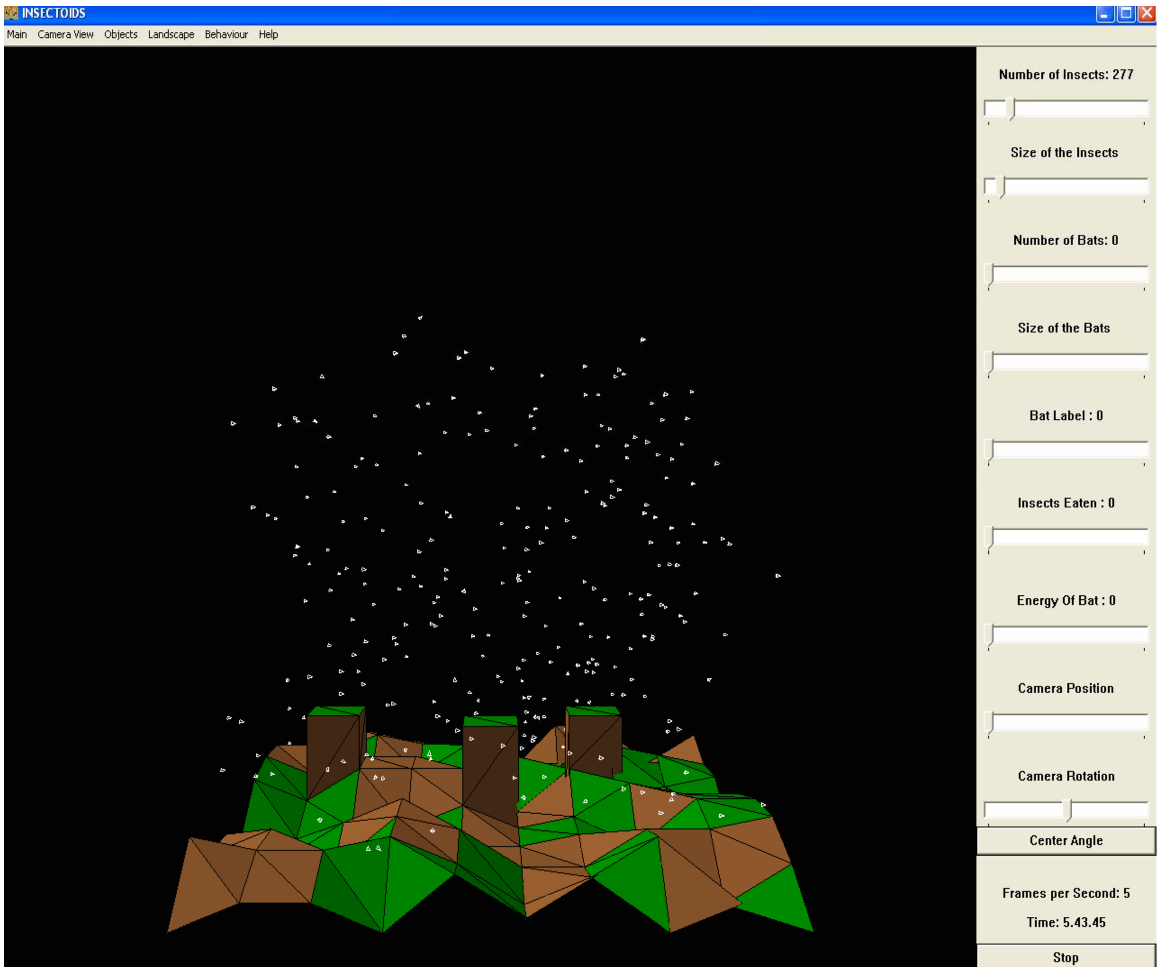

Figure 3.10: Snapshot of collision avoidance rule.

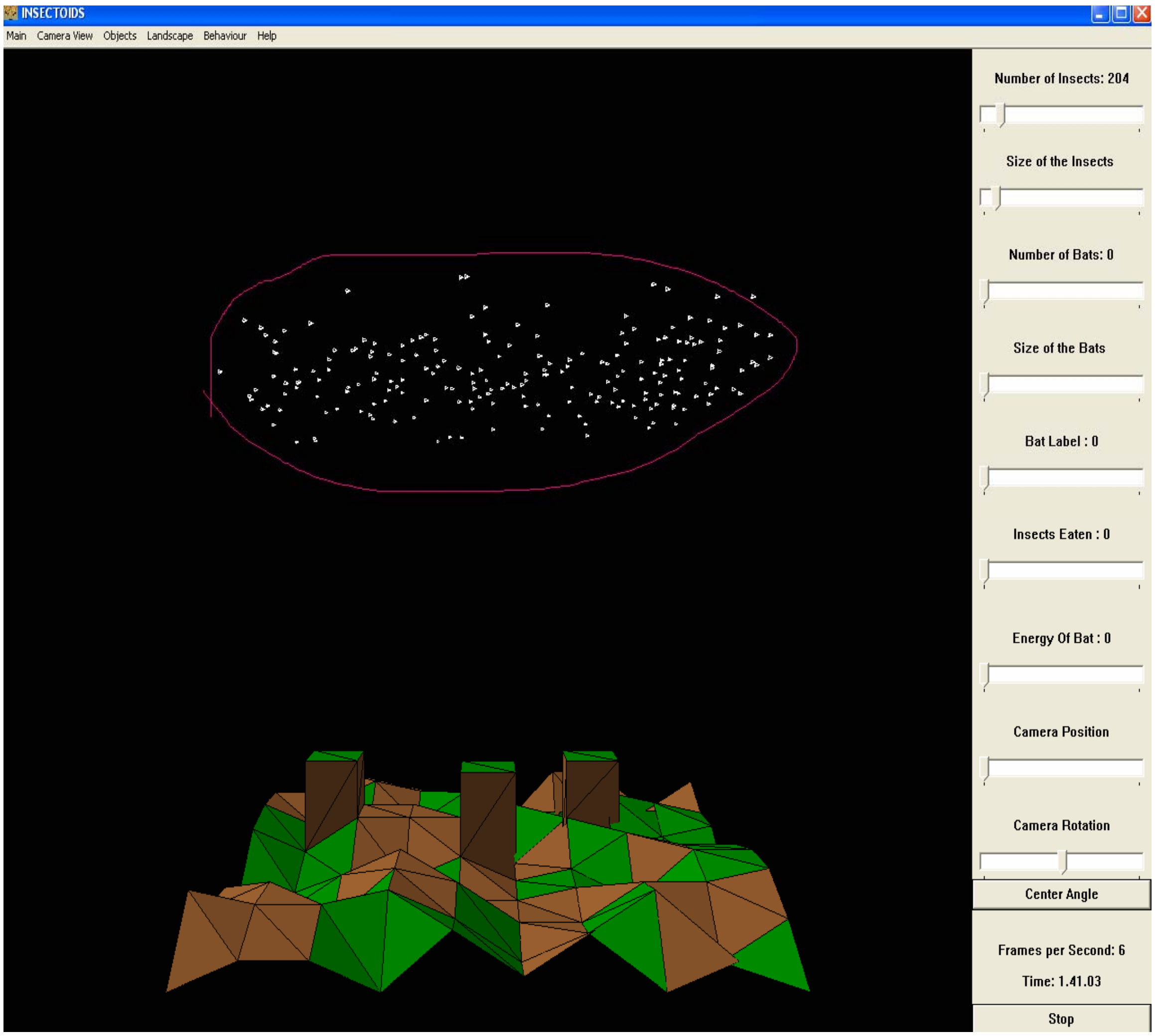

Figure 3.11: Snapshot of the insects migration rule.

Data: i_i (an individual insect)

Result: updated position of the insect i_i after applying the insects landing rule.

```
FOR each insect i_i
        IF y co-ordinate of i_i  greater than 400
        y co-ordinate of i_i is assigned a random value less than 400
              END IF
END FOR
```

Figure 3.12: Pseudo code for insects foraging rule.

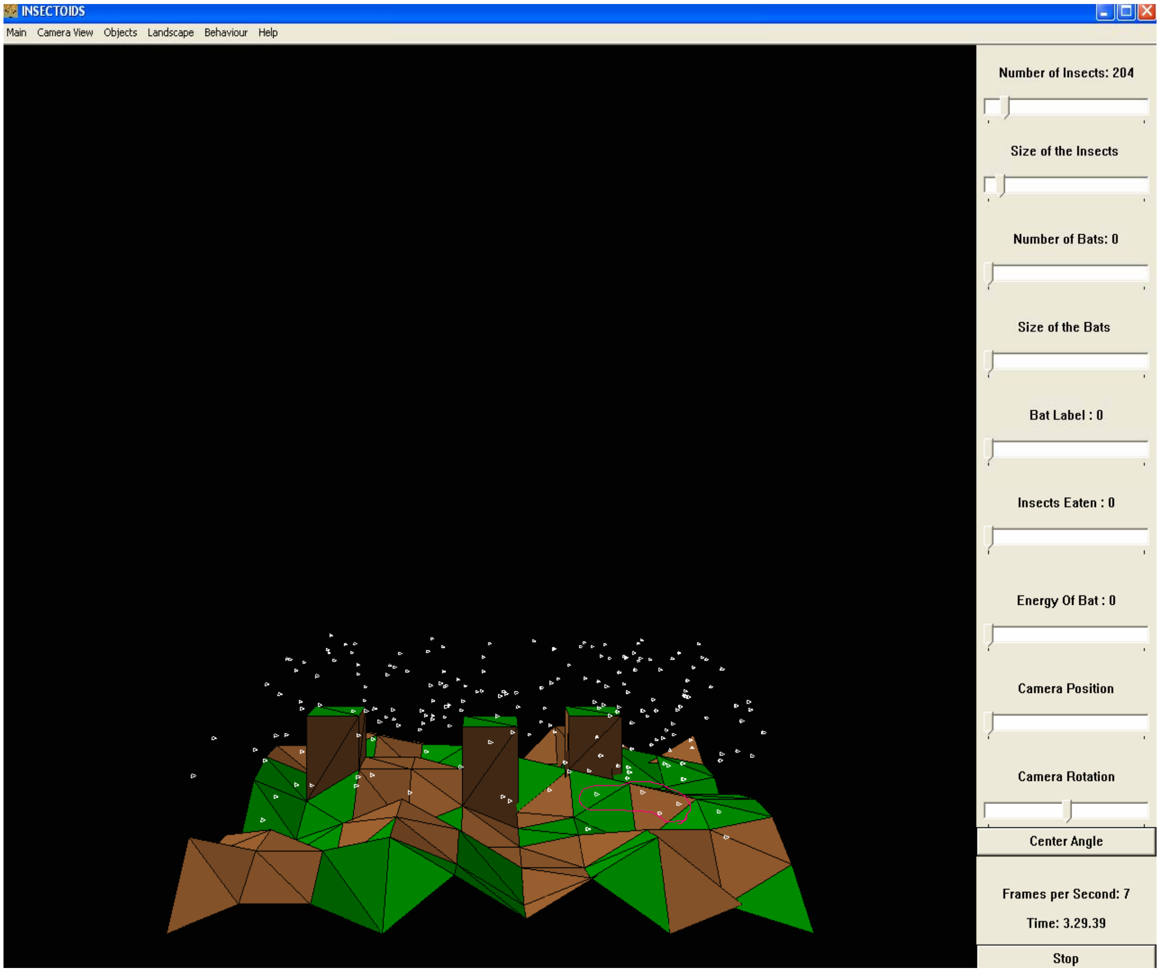

Figure 3.13: Snap shot of insects foraging rule.

3.3 Simulator Interface

Using Visual C++ the simulator has been designed to be user friendly. Figure 3.14 shows the entire view of the simulation environment. The 3 dimensional space visualized cannot be measured in standard units of length. Each menu in the simulator will be explained in detail here.

3.3.1 Main Menu

This menu is used to navigate through the simulator. It has 3 submenus. Figure 3.15 shows the submenu.

- “Model” is used to specify the movement(s) to be simulated. If *INSECTOIDS* is selected the movement of insects is simulated. If *BATOIDS* is selected the movement of bats is simulated. If *COMBINED* is selected the movement of bats and insects are simulated. The *COMBINED* model is used to study the interaction between the bats and insects.
- Stop is used to pause the simulation and Start is used to resume the simulation.
- Exit is used to stop and close the application.

3.3.2 Camera View Menu

This menu is used to control the position of the camera. This allows better view of movement of the bats and insects. Figure 3.16 shows the different submenus present in the camera menu.

- Looking North: This is the view seen facing the north direction.
- Looking South: This is the view seen facing the south direction.
- Looking East: This is the view seen facing the east direction.

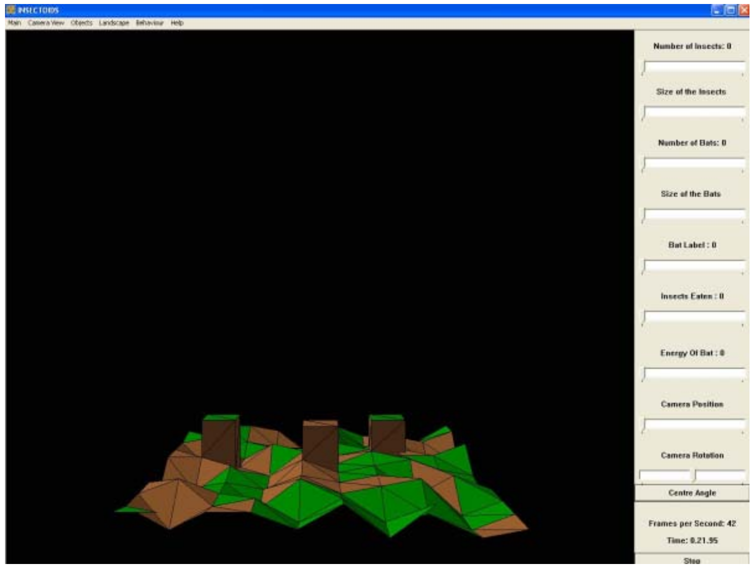

Figure 3.14: Simulation entire view.

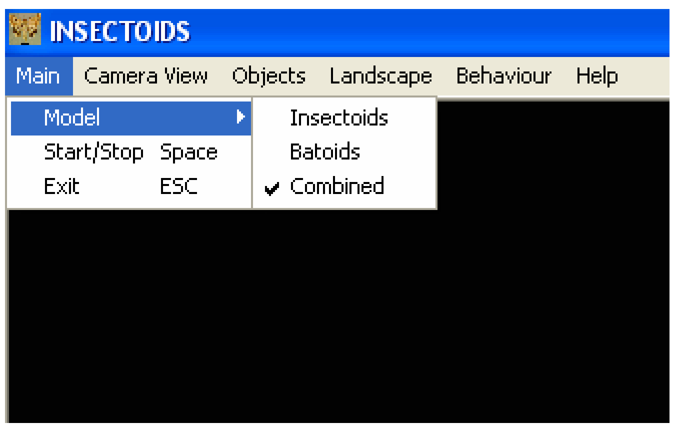

Figure 3.15: Main submenu.

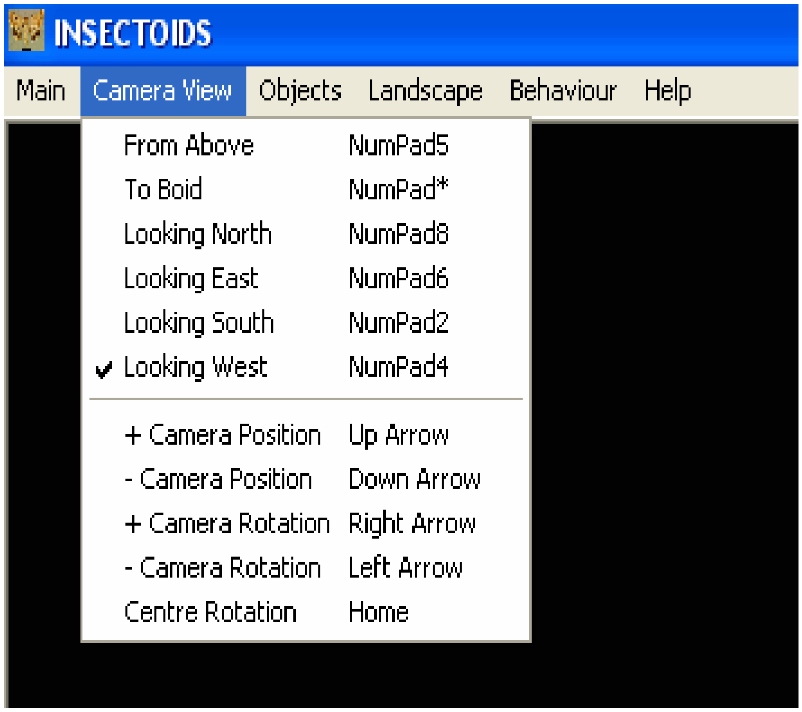

Figure 3.16: Camera view submenu.

- Looking West: This is the view seen facing the west direction.
- To Boid: In this view the camera is placed on the current bat on which iteration is being done. It gives a clear picture of the movement of each bat from an individual bats perspective relative to the rest of bats.
- + Camera Position: This is used to zoom in the camera to have a closer view.
- -Camera Position: This is used to zoom out the camera to have larger view.
- Camera Rotation: This is used to rotate the camera according to user's requirement. It improves the 3D visualization of the entire scenario.

3.3.3 Objects Menu

This menu is used to change the color and mesh used for the insects and bats. Figure 3.17 shows the submenu of object menu. The submenu of Objects prescribes appearance of the simulated objects (bats or insects) but has nothing to do with the movement of the bats and insects.

- The wire frame, unlit flat, flat and gouraud menus are used to change the geometry of the mesh used for the insects and bats. In wire frame the mesh is made of wire. While in the case of other meshes, the whole mesh is filled with color, which hides the structure of the mesh.
- Add Flyer and Remove Flyer: This is used to add or remove bats from the simulator. This can also done using the slide bar.
- Add Insects and Remove Insects: This is used to add or remove insects from the simulator. This can also be done using the sliding bar.
- Bats Color: This is used to change the mesh color of the bats. We require different colors for the bats and insects to identify them in the simulation.

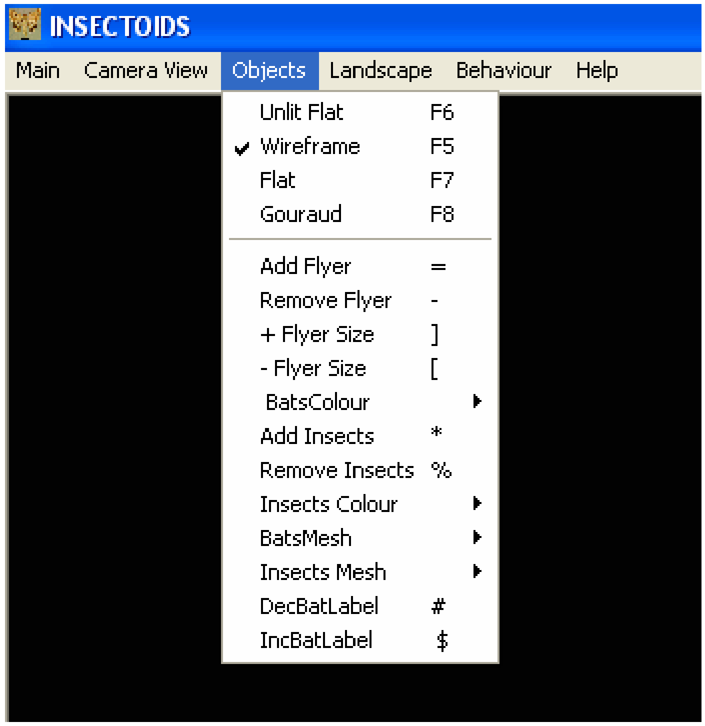

Figure 3.17: Objects submenu.

- The colors that can be given to bats are white, pink, and yellow. The default colors used for the bats are yellow.
- Bats and Insects Mesh: This is used to change the mesh of the bats and insects. The meshes available are tetrahedron, small bird. The default mesh used for bats are small bird and the mesh used for insects is tetrahedron.
- +Flyer Size and –Flyer Size: This is used to increase and decrease the size of the bat. This can also be done using the sliding bar present on the right in the simulator.
- Inc and Dec bat label: This is used to increment or decrement the label of the bat to obtain details regarding the energy spent and insects eaten by that specific bat. Only using this, the bat label can be modified in the sliding bar.

3.3.4 Landscape Menu

Using this menu the landscape for the bats and insects can be changed. The different options available are wire frame, Unlitflat, flat and gouraud. Using "recalculate" the position of the green and brown patches can be changed. The green patch denotes crop area and the brown patch denotes non-grassy area. The position of the green and brown patches is not fixed. Each time the simulator is opened, they are randomly placed. Using solid coloring the landscape color can also be changed. Sometimes we can also remove the landscape to reduce computational load on graphics using the "none" option. Figure 3.18 shows the landscape menu.

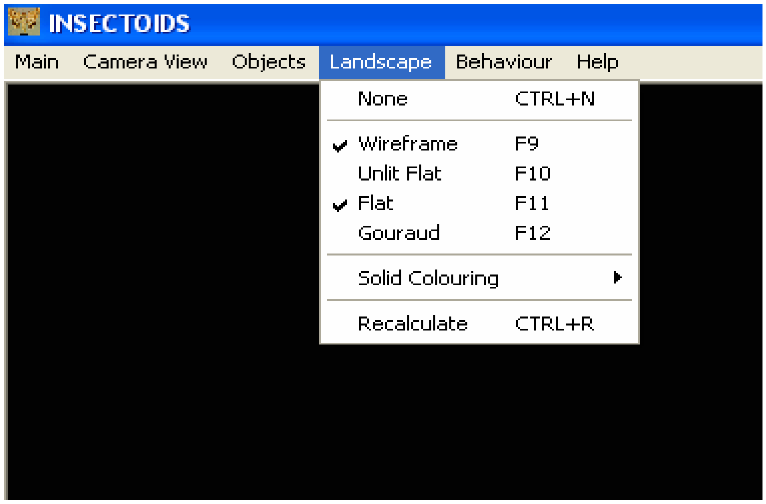

Figure 3.18: Landscape Submenu.

3.3.5 Behavior Menu

This menu controls the movement of the bats and insects depending on the rules selected in their respective submenus. Figure 3.19 shows the submenu of behavior menu. This is the major area where this thesis makes a contribution.

- Insect Attributes: This is a dialog box used to specify the attributes of the insect. The attributes are maximum and minimum speed, probability of insect escape, acceleration rate, angle of vision, collision distance, landscape collision distance, landing speed, probability of insect catching, and starting speed.
- Insect Rules: These are rules which affect the motion of the insects. They can be implemented for the insects when the model selected is *INSECTOIDS* or *COMBINED*. The rules are collision avoidance, insect migrating and insect landing.
- Bat Rules: These are rules that are implemented for the movement of bats when selected. The primary rules are collision avoidance and individual predator avoidance. The remaining rules are secondary. These rules can be implemented when the model is either in *BATOIDS* or *COMBINED* mode.
- Bat Attributes: This is a dialog box used to specify the attributes of the bat. The attributes are acceleration rate, angle of vision, collision distance, flock forming distance, flocking radius, distance of detection,

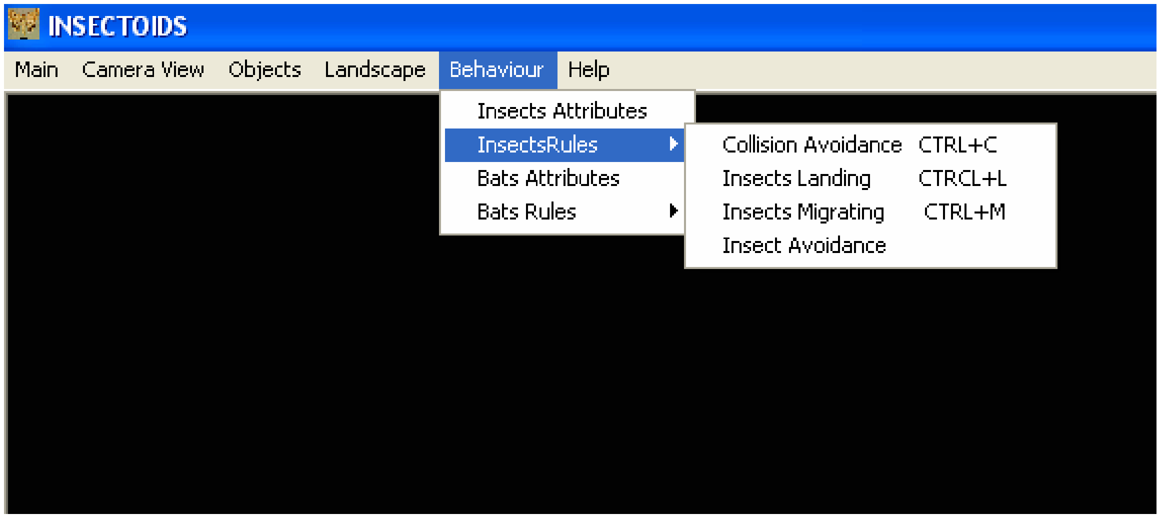

Figure 3.19: Behavior Submenu.

angle of elevation, bats energy reserve, minimum speed, maximum speed, range of flock headings and emergent speed of bat.

3.3.6 Control Bars

The different parameters that can be controlled using the controls bar present on right side of the simulation are:

- The number of bats and insects present in the scene.
- The size of the insects and bat present in the scene.
- The label of the bat can be increased or decreased according to requirement. Depending on the bats label the corresponding information of insects eaten and energy is displayed.
- Using the camera rotation button the camera position can be changed.
- Camera can be – zoom in or zoom out.
- The stop/start button is used to pause and restart the simulation.
- The number of frames indicates whether the simulation is overloaded. If the number of frames per second is less than 3, the simulation is overloaded and automatically some insects and bats are removed from the simulation.
- The time display indicates the total time for which the simulation is running. Figure 3.20 is a snapshot of the control bar.

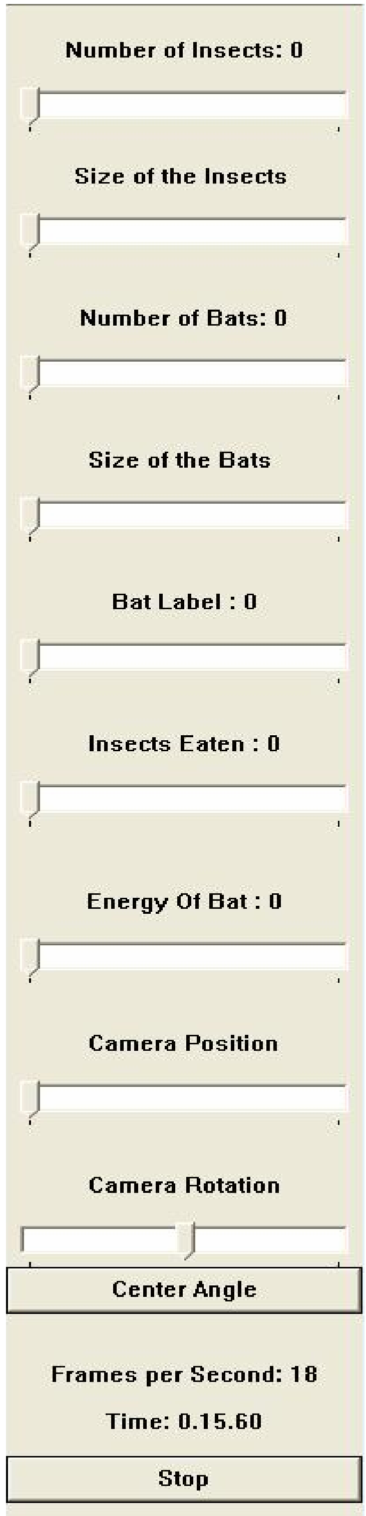

Figure 3.20: Control Bars.

CHAPTER 4

Experiments and Results

4.1 Foraging Pattern of Bats for Constant Insect Density

There are many behavioral components (e.g. prey selection, habitat selection, and determination of the time of foraging and duration of feeding bouts) that affect foraging habits of bats. They can be modified to maintain maximal energetic efficiency under varying environmental conditions. Conditions that may influence the time and duration of foraging flights and night roosting periods are: 1) temporal aspects of prey activity 2) prey abundance 3) predator activity 4) energetic constraints (Schoener 1971). Due to a relatively high cost of flight, the metabolic expense of foraging in bats is great (Thomas 1975). Therefore, bats must have substantial amount of energy to survive even after performing tasks like feeding and then roosting. When insect density is low or cool temperatures prevail, bats spend less time foraging and more time roosting (Anthony, Stack et al. 1981). Therefore bats cease foraging when poor foraging success and/or high costs of flight and thermoregulation prevent maintenance of a positive energy balance. When foraging is successful bats return to their night roost due to satiation (Anthony, Stack et al. 1981).

To study the foraging patterns of the bats in the field, we introduced up to 5 bats in the simulation in the presence of varying constant insect densities. The initial positions of the insects were random. For every insect eaten by a bat according to the pursuit and capture rule, a new insect was generated. This way the insect density was maintained at a constant value. Insects eaten by the bats depend on the availability. If the insect density is high, the number of insects eaten is also high and vice-versa. In Figure 4.1, we have plotted the insects eaten by 5 bats with insect density varying from 10-200 in the steps of 50. It is evident from the graph that insect consumption has increased with increasing insect density. If a bat eats more insects they spend more time in the field foraging. Foraging time is the time difference between flight end and first insect eaten times. Figure 4.2 shows the foraging time spend by the bats with increasing insect density. It can be observed that foraging time was less when insect density is less. It has increased with increasing insect density but there is a decrease at 200 insect density level as the bat returns earlier due to satiation. Commuting time was calculated as the difference between start time and first insect eaten times. Figure 4.3 shows the commuting and foraging time spent by the 5 bats in the presence of varying insect density.

Day-to-day variations in costs of flight and thermoregulation as well as seasonal changes in energy demands of reproduction make energetic considerations for bats complex (Anthony, Stack et al. 1981). In the simulation, energy balance of the bat was calculated taking into account the energy spent by the bat in flight by foraging, commuting, maintaining it's metabolism, roosting and energy gained by feeding on insects .

*Energy Balance of Bat = Initial energy of bats + Energy gained eating per insect * Number of insects eaten – Energy spent per second in flight * total time of flight - Energy spent in maintaining its metabolism – Energy spent in roosting.*

Energy obtained by feeding on single *H.zea* moth is approximately 3.29 kilo joules (Bushman, McGinleyB et al. 2002). In the simulation initial energy of the bats is 200 kilo joules.

Using the models proposed by Norberg and Pennycuick (Pennycuick 1989, U.M. Norberg et al. 1993), the energy spent by bat in flight is estimated to be 30 joules per second . Energy spent per second for maintaining metabolism is 264.63 joules (Kunz 1982). The energy spent in roosting is calculated using the following formula:

Energy spent in roosting = exp (1.6317 + 0.719*log (mass) - 0.0187 * roostTime)*0.01998

where roostTime is obtained by subtracting flight time from 24.

Figure 4.4 shows the energy balance of the bats for varying insect density. It can be seen that when the insect density is less the bats return to the cave with an energy less than initial energy reserve. With increasing insect density due to high capture rates the bats return with energy higher than initial energy.

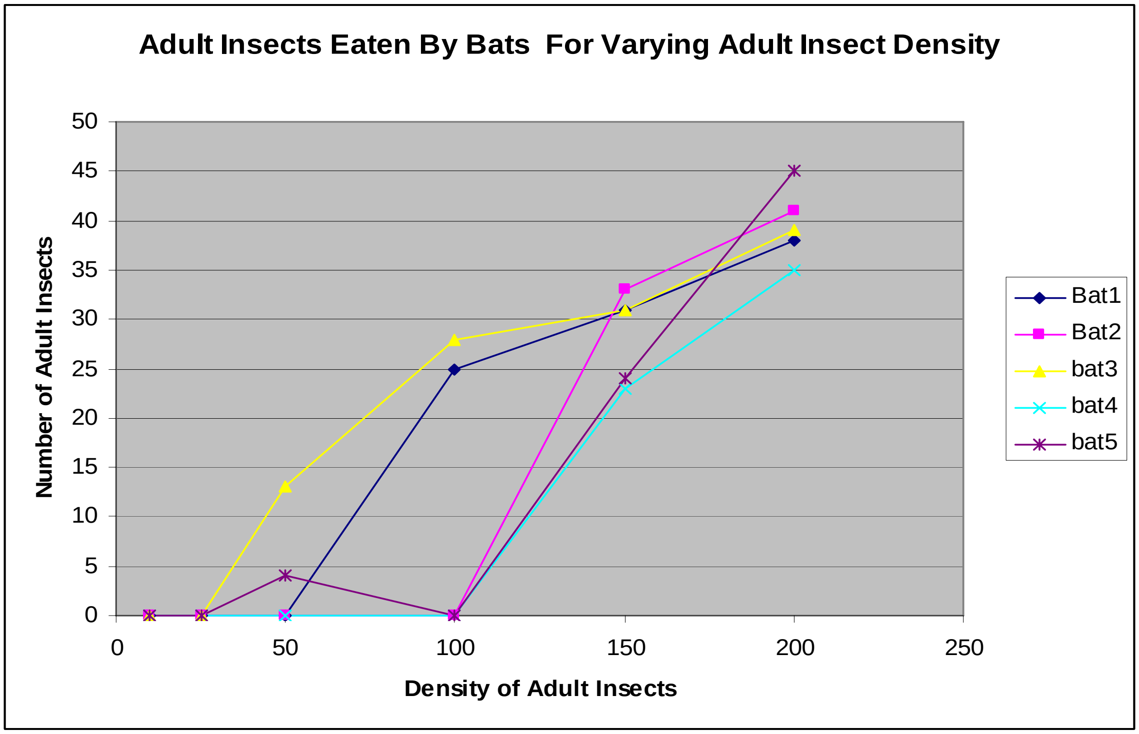

Figure 4.1 Insects eaten by 5 bats for varying insect density.

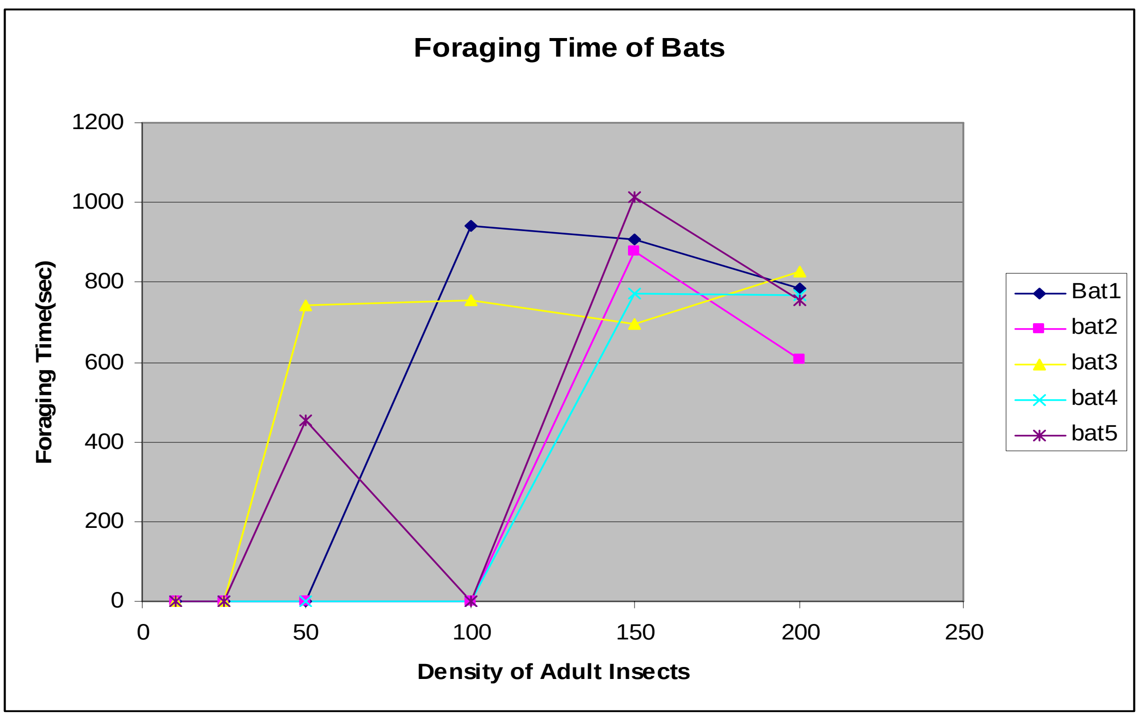

Figure 4.2: Foraging time of 5 bats for varying insect density.

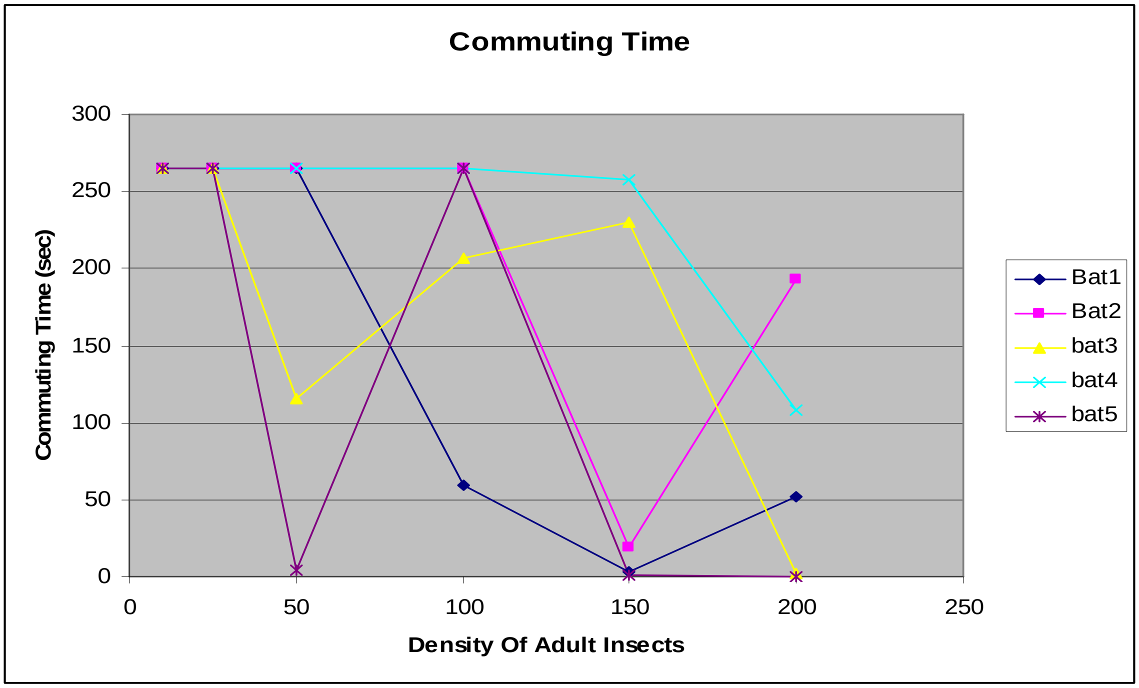

Figure 4.3: Commuting time of 5 bats for varying insect density.

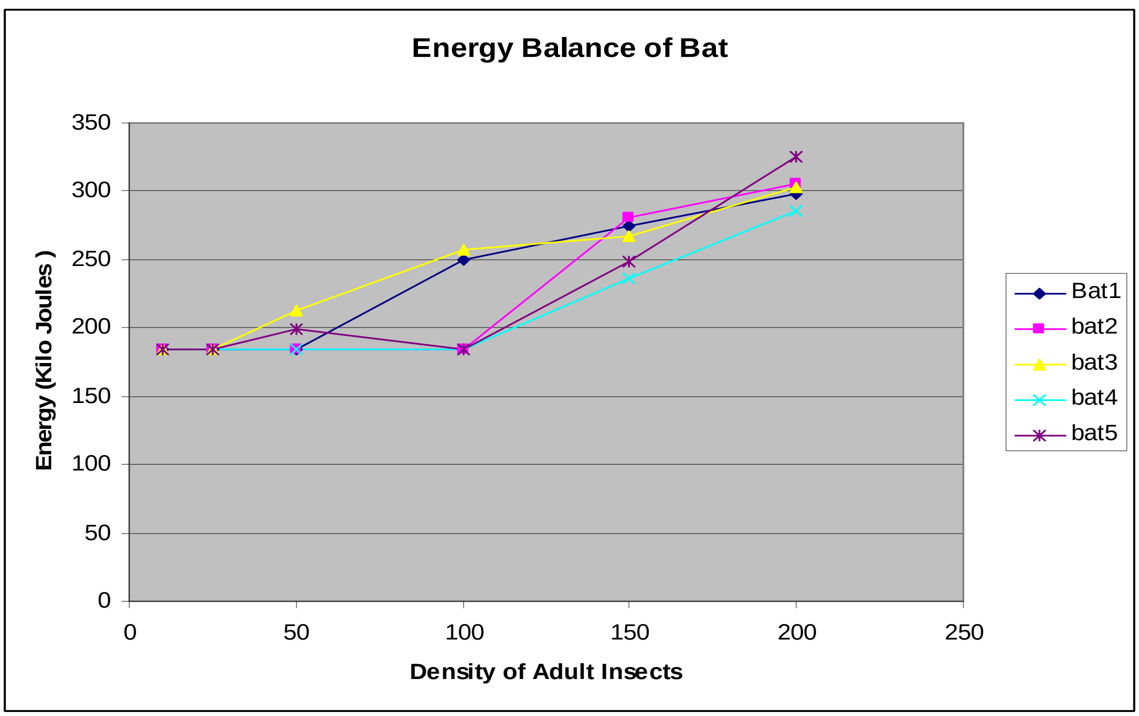

Figure 4.4: Energy balance of 5 bats at the time of return.

4.2 Single Bat Behavior In The Case Of Step Up and Step Down Of Insect Abundance

A single bat was introduced in the simulator in the presence of insect density levels of 200 and 100. Implementing the pursuit and bat saturation rules for the bat, the details regarding flight time, insects eaten and foraging time of the bat were recorded. Table 1 shows the values recorded. After the bat feeds on half of the regular amount of insects eaten at each insect density, the insect density was increased from 100 to 200 and decreased from 200 to 100. Table 2 shows the various details recorded for the bat. Comparing the tables, we can see that when insect density was increased from 100 to 200 the bat ate more insects and spent less foraging time due to the increase in insect density. The percentage increase in insects eaten is 77.27%. Whereas when the insect density was decreased from 200 to 100 the bat spent more time foraging the field, eating less insects. The percentage decrease in insects eaten is 20%. The percentage increase/decrease validates that insect density is directly proportional to the insects eaten by bats.

4.3 Varying Efficiency of the Bat to Capture Insects

The interactions between bats and insects are in a category that has often been termed “a coevolutionary arms race”. It has been demonstrated that insects have auditory systems adapted to the echolocation system of bats that prey on them, and that bats, in return, have altered their echolocation calls and/or foraging behavior to overcome the insect’s defenses(Waters 2003).

Table 1: Details recorded for a single bat with constant moth density.

Insect Density	Insects Eaten	Flight Time	Foraging Time
100	22	1040	873.681
200	35	876	855.681

Table 2: Details recorded for a single bat with a change in moth density.

Insect Density	Insects Eaten	Flight Time	Foraging Time
Step Up 100-200	39	824	715.312
Step Down 200-100	28	970	902.86

In this regard, to study the foraging pattern of bats, simulations were run with varying probabilities of insect capture for constant insect density of 200. The efficiency of a bat is a measurement of bats capability to catch the insect within its cone of detection. When the efficiency of bats is high, it is more capable to capture and feed on the insects within its cone of detection. In the simulation the efficiency of bats is measured with values from 0, which corresponds to the least efficient bat in foraging capability to 1.0, which corresponds to highest efficient bat. Figure 4.5 shows the insects eaten and time spent by the bat outside the roost. It is clear that as efficiency of the bat to capture insect increases, the number of insects eaten have increased. With increasing efficiency the bats spent less time outside the roost. This is evident from Figure 4.6.

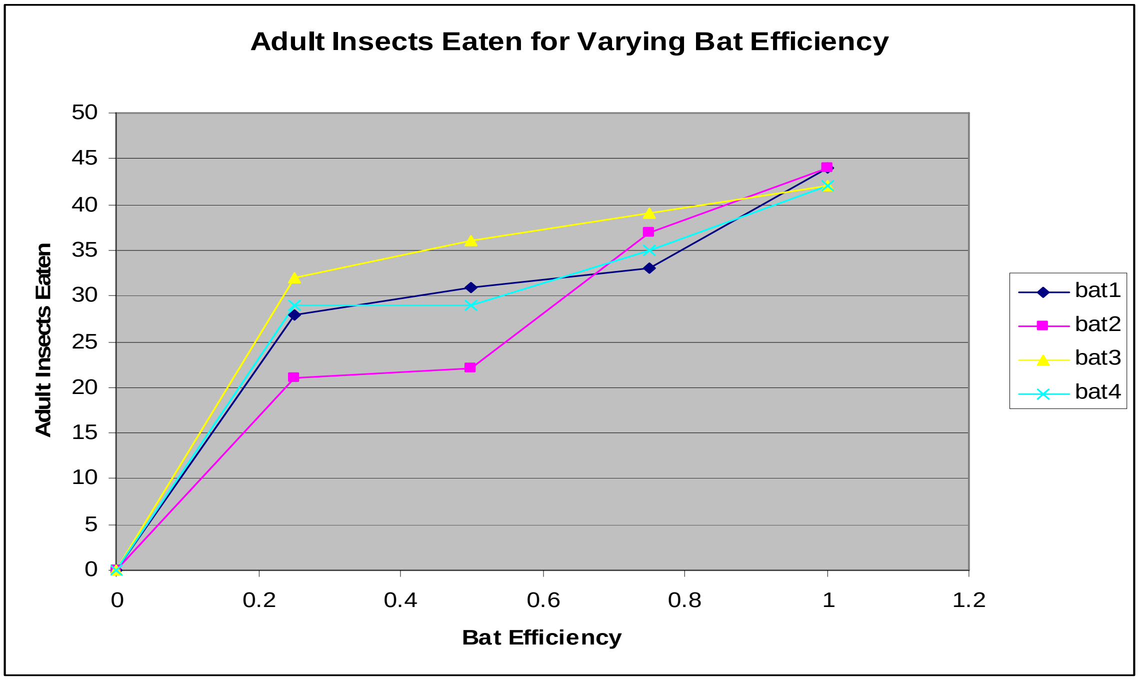

Figure 4.5: Insects eaten for varying successful foraging rates of bats.

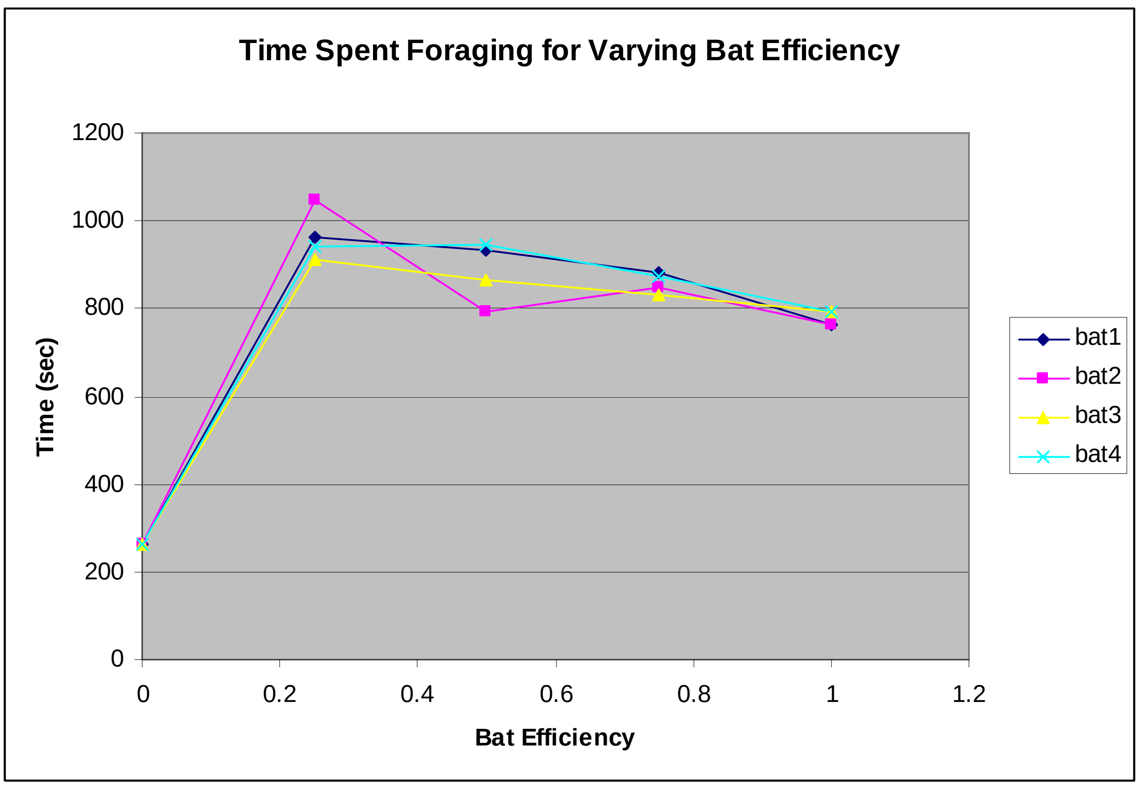

Figure 4.6: Time spent foraging for varying bat efficiencies.

4.4 Varying Escape Response of Insects

Insects have the ability to detect the bats calls and exhibit an escape response (Waters 2003). Acharya and Fenton (1999) demonstrated that insects exhibiting escape behavior were caught significantly less than those that did not. (Roeder 1962) also found that only 13% of the insects that did show escape behavior were caught and 87% of insects that failed to exhibit escape maneuvers were caught. (Rydell J., N. Skals, et al. 1997) reported that insects that detected bats at distances less than 5m tended to spiral or dive to the ground, whereas insects that detected bats at distances greater than 5m changed their path. To study the foraging pattern of bats when insects exhibit escape response, we coded the insect's model to have a high efficiency to escape, by forcing it to fly down. By doing so the insect is outside of the cone of detection and the bat doesn't capture it. By the term efficiency of insect, we refer to its ability to escape from being captured by the bat. The insect's efficiency was given values between 0 to 1. 0 where 0 refers to lowest efficiency and 1 refers to high efficiency of the insect to exhibit escape behavior. Figure 4.7 shows the number of insects eaten by the bat as a function of escape response by moths. It can be seen that the number of insects eaten by a bat decreases with an increase in the efficiency of the moth to escape. Figure 4.8 shows the foraging time of bats in search of insects that exhibit escape response. Time spent by a bat foraging also decreases with an increase in efficiency of insect to escape capture. It is beneficial for the bat to save energy by roosting rather than foraging when success rate is low.

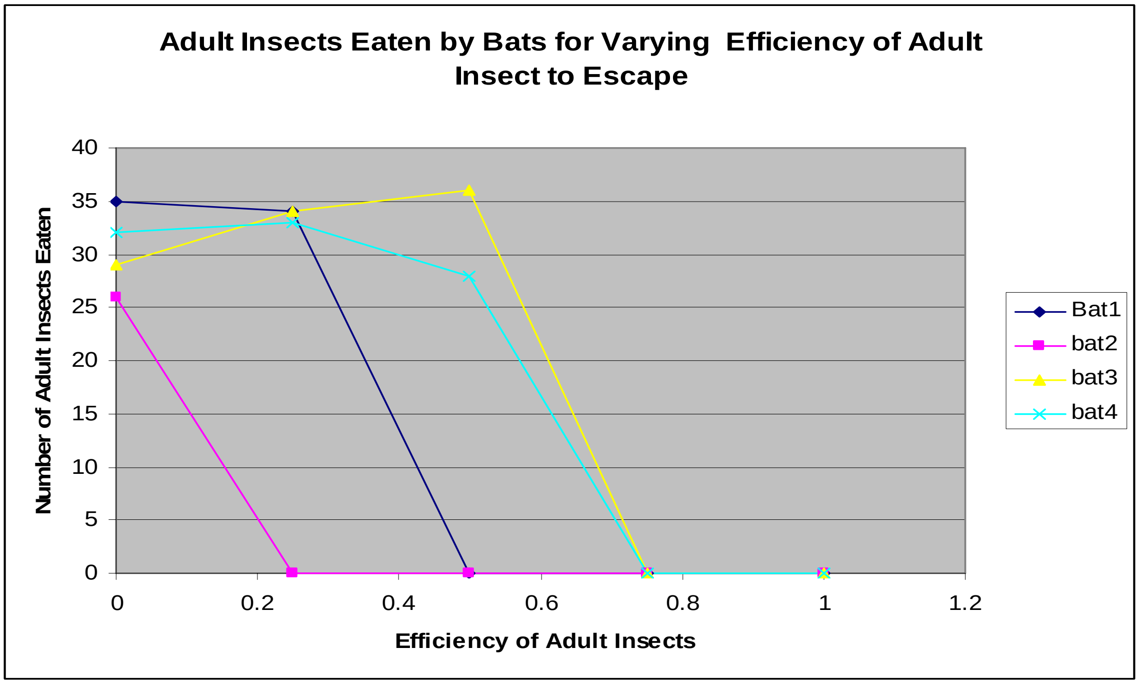

Figure 4.7: Insects eaten by bat as a function of escape response by insects.

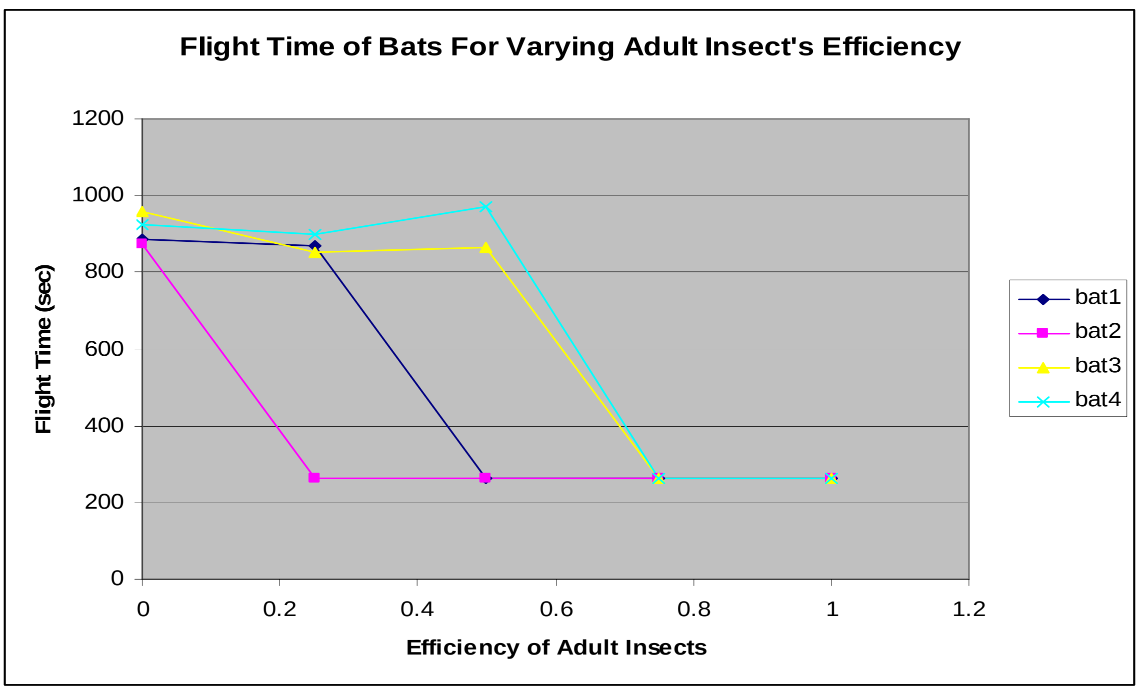

Figure 4.8: Flight time of bats for varying escape response by insects.

CHAPTER 5

Moth Counting

Traditionally, insect pest populations were controlled by applying broad spectrum synthetic pesticides that can be dangerous to human health and the environment. With the isolation of microbial toxins from the soil bacterium (Bacillus thuringiensis) (*Bt*), sprays have become available that are organically derived (Jorge Fernandez-Cornejo and William. D. M. 2000). The genes of *B.thuringiensis* have also been incorporated into the genome of numerous agricultural crops to target specific taxa (Lepidopteran larvae). The use of *Bt* crops has become widespread because of the benefits such as fewer broad-spectrum pesticide applications and increase yield for growers. The use of transgenic crops, such as those producing Bt (*B.thuringinesis*) insecticides, raises concerns that non target species may be negatively impacted and food webs disrupted (Marvier 2001).

In this chapter I propose a technique to count the moths present in videos recorded in *Bt* cotton fields at Texas. By estimating the number of moths in the *Bt* cotton fields, we can investigate the affects of *Bt* crops on moth densities and subsequently on bats foraging activity. The coding for the application was done using C#.net.

5.1 Need for Automation

Counting moths manually observing the video is very tiresome. At normal speed there is possibility of losing a moth even at blink of our eye. To be accurate we have to reduce the speed of the video as the moths are seen for few seconds (nearly 2 -3 seconds). Depending on the distance of the moth from the camera its intensity varies. The highest intensity of the moth seen in the videos recorded was 40. And the background intensity is 10 in the videos. As it is hard to detect the moths from the background because of this low difference, we have to increase the contrast in the original video. To solve this problem automating the counting process was performed using Image segmentation. Image segmentation techniques allow to distinguish the required object from the background objects.

5.2 Camera Setup

Sony DCR-TRV11 digital video cameras were mounted at a height of 1.5m in corn plots (0.5 m in corn plots) and aimed upward with the top of the camera view pointed north. The cameras were focused using a test pattern target at a range of 2.5m where the camera had approximately a 1.3 m X 1.7 m viewing area (Fig 5.1). Cameras were operated in NightShot mode at a simulated shutter speed of 1/30 seconds. Long play mode was used to extend the tape recording time to 2 hours. Infrared lights were placed 2m to either side of each camera and pointed to intersect at a height of 2.5m above the camera. The Infrared lights were powered by a 12Volts DC battery and pulsed by a controller circuit for 4 ms at a rate of 60 Hz to enhance the illumination of targets.

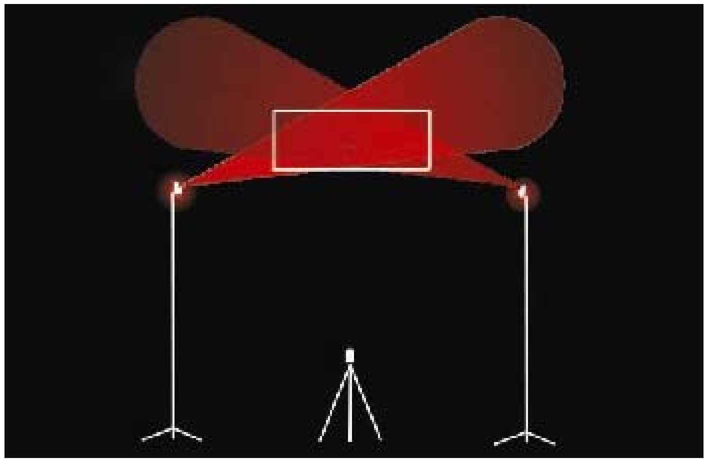

Figure 5.1 Schematic Diagram of Camera Setup.

5.3 Image Segmentation

Segmentation involves distinguishing an object from background. The goal of segmentation is to change the representation of an image to facilitate further analysis. Several general purpose algorithms have been developed. The most relevant for our problem are thresholding and connected components labeling. They are based on partitioning an image into regions that are similar according to a set of pre-defined criteria.

5.3.1 Thresholding

Thresholding segments an image by setting all pixels whose intensity values are above a threshold to a foreground value and all the remaining pixels to a background value. Two thresholding techniques have been used - Fixed thresholding and automatic thresholding. The threshold value in fixed thresholding was determined using the histogram. From the histogram of the image, we can observe that pixels corresponding to a moth range from 15–40 depending on the distance of the moth from the cameras. Pixel intensity of the background pixels is 10. Figure 5.2 and 5.3 is the histogram of the frame with a moth and without a moth. In fixed thresholding, if pixel intensity is greater than average pixel intensity by 10, they are considered foreground pixels otherwise they are background pixels. Figure 5.4 is the frame in which moths is there before performing thresholding. Figure 5.5 is the same frame obtained after performing thresholding. To determine the threshold automatically – iterative optimal threshold selective algorithm (#2) has been used.

The step-by-step iterative optimal threshold selective algorithm is given below.

1 Consider as a first approximation that the four corners of the image contain background pixels only and the remainder contains object pixels.

2 At step n, compute the mean background ($\mu^n{}_b$) and object gray level ($\mu^n{}_o$) where threshold T^n (determined in the previous step) defines segmentation into background and objects.

3 Set $T^{(n+1)} = (\mu^n{}_b + \mu^n{}_o)/2$

  $T^{(n+1)}$ is the updated threshold value.

4 If $T^{(n+1)} = T^n$, Stop, otherwise return to step2.

5.3.2 Connected Components Labeling

Connected components labeling groups pixels in an image into components such that all pixels in a connected component have the same pixel intensity and are connected with each other. The classical labeling approach (Rosenfeld and Pfaltz 1996) performs two raster scans of the image. The step-by- step 4 – connected component labeling algorithm is given below:

1. Scan through each pixel $P_{(i,j)}$ in the image. Assign labels to $P_{(i,j)}$ according to the label of the pixel above it $P_{(i-1,\, j)}$ and the label of the pixel just in front of it $P_{(i,\, j-1)}$.

   If $P_{(i,j)}$ is a foreground pixel, and neither $P_{(i,\, j-1)}$ nor $P_{(i-1,j)}$ is labeled , then $P_{(i,j)}$ is assigned a new label.

   If either $P_{(i,\, j-1)}$ or $P_{(i-1,j)}$ is a foreground pixel , then $P_{(i,j)}$ is assigned either $P_{(i,j-1)}$'s or $P_{(i-1,j)}$'s label.

   If both $P_{(i,\, j-1)}$ or $P_{(i-1,j)}$ are foreground pixels, then $P_{(i,j)}$ is assigned the smallest label among $P_{(i,\, j-1)}$ or $P_{(i-1,j)}$ and note equivalences if any.

2. Rescan to consolidate equivalent labels.

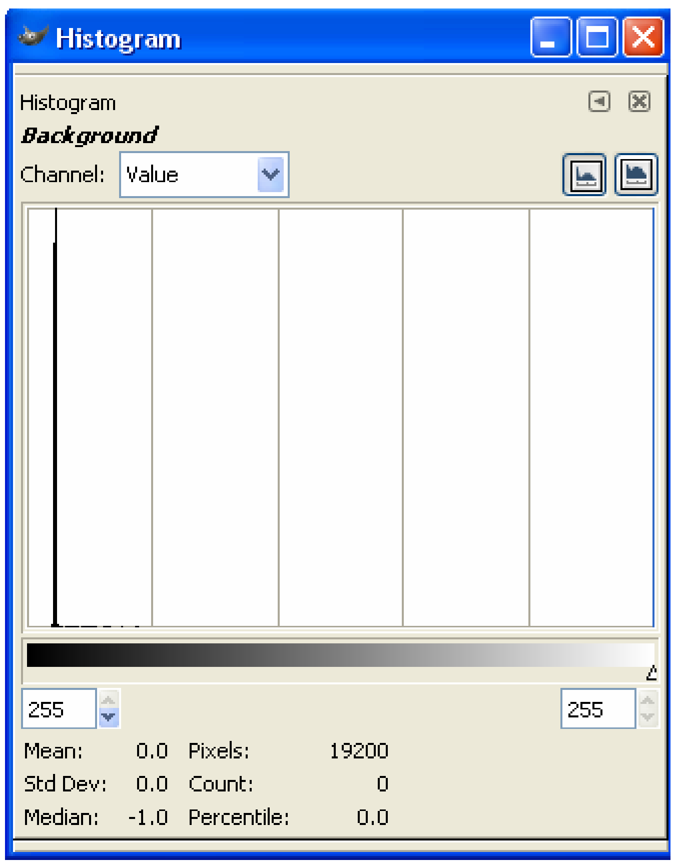

Figure 5.2 Histogram of a frame with a moth.

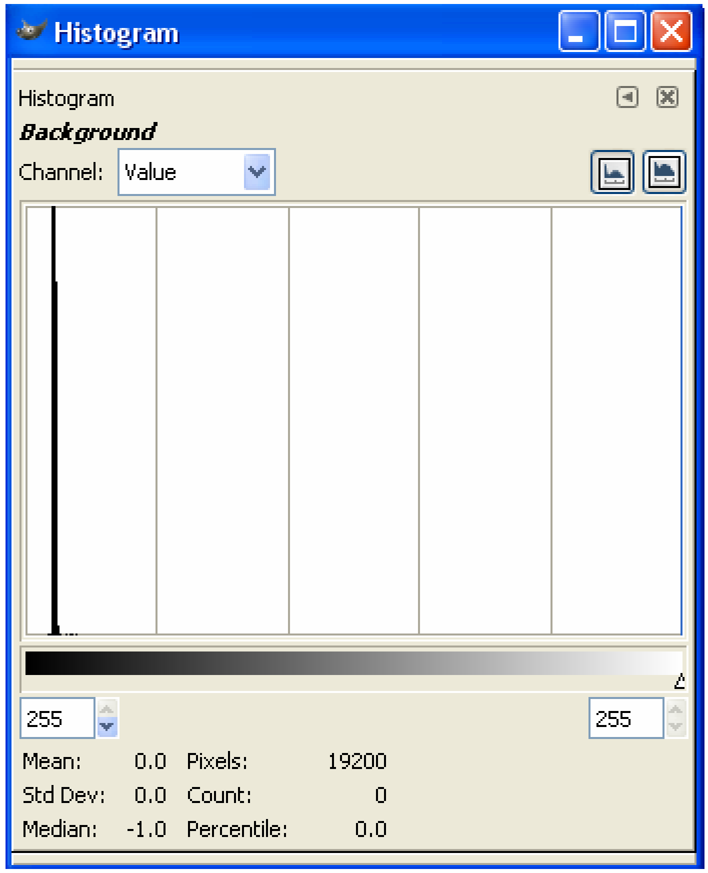

Figure 5.3 Histogram of a frame without a moth.


Figure 5.4: Frame 281 before thresholding

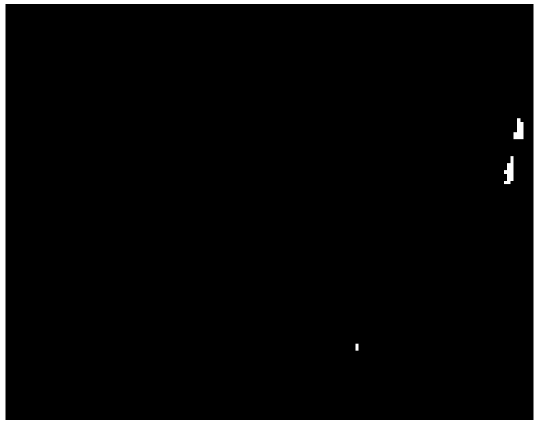

Figure 5.5 Frame 281 After Thresholding.

5.3.3 Moth Counting Algorithm

To enable counting the moths, the frames in which moths were present were pulled out from the video and saved in a local directory as bitmap image. Stepping through all the bitmap images we can count the number of moths present in the video recorded.

The step-by-step algorithm to determine the presence of a moth in the frame:

- Threshold the image.
- Label the foreground pixels in the threshold image.
- If the number of foreground pixels greater than 2 and less than 300. Save the image.
- Else neglect the current image and repeat steps 1-3 on the next frame in the video.

5.4 Interface

The moth counting algorithm was developed in C#.Net. The inputs required are -path of the video file and the number of frames per second at which video was recorded. To track the progress of the frames analyzed in the video file, information regarding the current frame analyzed and the total number of frames in the video are displayed. The frames which have moths are saved in the file path of the video file. Stepping through these frames we can estimate the number of moths in the video. Figure 5.6 is a snapshot of the interface developed.

Using this interface, counting moths becomes easier than observing the complete video manually. It takes lesser time to step through frames which have moths.

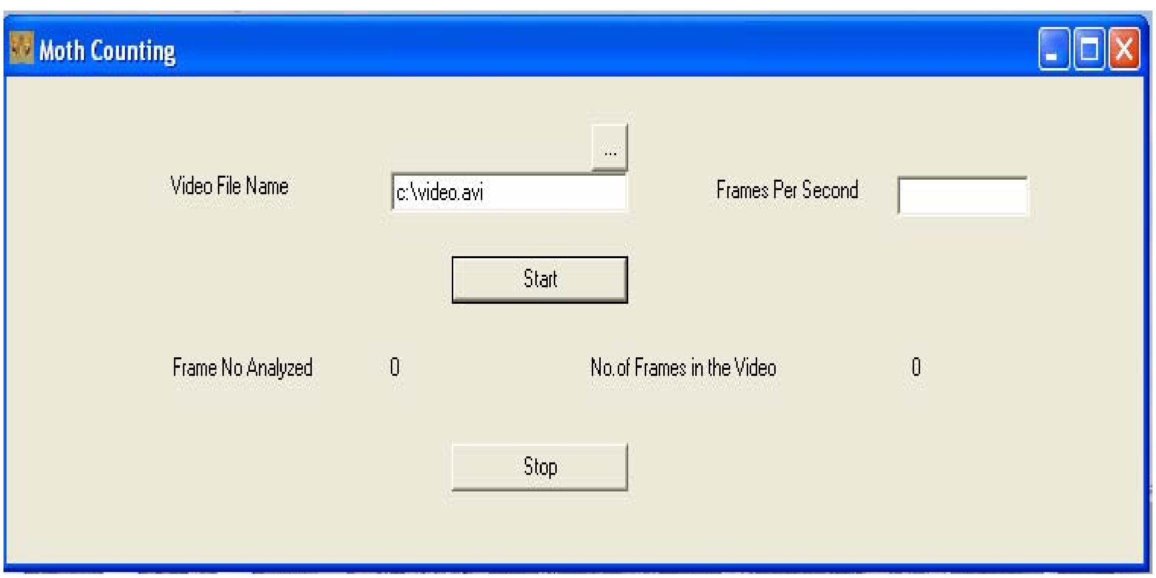

Figure 5.6 Moth Counting Interface.

5.5 Results and Discussions

The moth counting algorithms was applied on 3 video files of different lengths recorded at the field. For each video file both fixed and automatic thresholding were applied. To count the number of moths present in each video we stepped through the frames saved in the directory in which the video file was present. Table 3 gives the count of moths in each video for fixed and automatic thresholding.

From the Table 3, we can see that 10% of the total frames were selected to detect moths in the video using moth counting algorithm and fixed thresholding technique where difference between current pixel value and average pixel intensity is greater than 5 was applied. 5.5 % of the total frames were selected to detect moths is the video using moth counting algorithm and fixed thresholding technique where difference between current pixel value and average pixel intensity is greater than 10 was applied. In the moth counting algorithm when automatic thresholding technique was used only 2% of the total frames were selected to detect moths. From the Table 3 we can see that by performing connected component labeling after thresholding we eliminated few unnecessary frames. These frames had one or two stars with no moths in it.

Moth counting using fixed thresholding technique is more accurate than automatic thresholding. Some moths that were too far away from the camera which had pixel intensity almost equal to background were not counted by neither fixed thresholding techniques. By lowering the threshold value there is a possibility to detect these moths. In automatic technique some moths

Table 3: Moth Counting Algorithm Results.

File Name	Algorithm	Moths in the Video	Moths Detected	Frames Selected After Thres-holding	Frames Selected After Labeling	Total Frames in the video	Accuracy
3 min video	Fixed Threshold – 5	14	13	312	207	2758	92.85%
3 min video	Fixed Threshold - 10	14	9	26	21	2758	64.85%
3 min video	Automatic Thresholding	14	7	16	16	2758	50%
15 min Video	Fixed Threshold – 5	37	33	3120	3052	26882	89.18%
15 min Video	Fixed Threshold - 10	37	28	4803	2223	26882	75.67%
15 min Video	Automatic Thresholding	37	17	633	236	26682	45.94%
1 hr Video	Fixed Threshold – 5	159	152	12680	11586	107942	95.59%
1 hr Video	Fixed Threshold - 10	159	136	17973	8577	107942	85.53%
1 hr Video	Automatic Thresholding	159	107	8791	4809	107942	65.40%

were missed because the threshold value determined by iterative optimal threshold selective algorithm was either too high or too low. The time required to implement the automatic thresholding algorithm is more than fixed thresholding technique. This is because for each frame using the iterative optimal threshold selective algorithm, threshold has to be determined.

CHAPTER 6

Conclusions

6.1 Contributions

The major contributions of the thesis work lie in the development of a simulator for study of bat and moths interaction along with associated energetics, and an automated image processing system for counting moths in real time video. We also simulate the movement of insects. The study of insect behavior is vital in understanding the foraging pattern of bats and facilitating the development of improved regional area-wide management and control strategies. Using the simulator, we investigated the effect of insect densities and their escape response on the foraging pattern of bats. By studying the interaction between bats and insects, we estimate the pest control provided by the bats in agro-ecosystems to help understand their ecological importance. The automated image processing system made moth counting in a real time video easier and faster. Image segmentation techniques such as thresholding and connected component labeling were utilized. Counting moths in these videos, we estimate the effects of Bt crops on moth densities and subsequently on bat foraging activity.

6.2 Future Considerations

The simulation allows only 2D visualization even though it is a 3D program. DirectX is used for simulating the graphics. To obtain 3D perspective we have to use different views available in the simulator. In the simulation, bats start foraging as soon as they are introduced in the scene. There is also a limit on the number of bats and moths that can introduced in the simulation. We can overcome these by using 3D game engine. By using a game engine, we can simulate the emergence pattern of bats and sub-flocking before they start to forage. By using this simulation we can estimate the total energetics of bats in commuting, foraging and roosting.

References

1 http://users.lmi.net/corben/anabat.htm#Overview.

2 http://iria.pku.edu.cn/~jiangm/courses/dip/html/node122.html.

3 Acharya, L. and M. B. Fenton (1999). "Bat attacks and moth defensive behaviour around street lights." Canadian journal of zoology **77**: 27-33.

4 Andrewartha, H. and L. Birch (1954). The Distribution and Abundance of Animals. Chicago, University of Chicago Press.

5 Anthony, E. L. P., M. H. Stack, et al. (1981). "Night roosting and the nocturnal time budget of the little brown bat, Myotis lucifugus: Effects of reproductive status, prey density, and environmental conditions." Oecologia **51**(2): 151-161.

6 Beerwinkle, K., J. Lopez, et al. (1995). "Annual patterns of aerial insect densities at altitudes from 500 to 2400 meters in east central texas indicated by continously operated vertically oriented radar." Southwestern Entomologist **18**: 63-79.

7 Beerwinkle, K., J. J. Lopez, et al. (1994). " Seasonal radar and meteorological observations associated with nocturnal insect flight at altitudes to 900 meters." Environmental Entomology **23**: 676-683.

8 Bushman, D. W., S. M. G. McGinleyB, et al. (2002). "The Role of Carbohydrate Metabolism in the Regulation of Postfeeding Diuresis in Adult Helicoverpa zea (Lepidoptera: Noctuidae)." Annals of the Entomological Society of America **95**(4): 492-497.

9 Cleveland, C. J., M. Betke, et al. (2006). "Economic value of the pest control service provided by Brazilian free-tailed bats in south-central Texas." Frontiers in Ecology and the Environment **4**(5): 238–243.

10 Council, W. H. (1999). "Bats (Order :Chiroptera)." Fish and WIldlife Habitat Management Leaflet **Number 5**.

11 Davis, R., C. Herreid, et al. (1962). "Mexican Free-Tailed Bats in Texas." Ecological Monographs.

12 Deangelis, D. L. and L. J. Gross (1992). Individual-Based Models and Approaches in Ecology: Populations, Communities and Ecosystems, Chapman & Hall.

13 Fenton, M. (1990). "The foraging behavior and ecology of animal-eating bats." J. CAN. ZOOL **68**: 411-422.

14 Fenton, M. B., P. Racey, et al. (1987). Recent Advances in the study of bats, Cambridge University Press**:** 152-170.

15 Hallam, T.G., Raghavan A., Dimitrov, D.T., Federico, P., and Qi, H., (2006) "Technology and Simulation of Bat Population Dynamics", *Proceedings of Environmental Modelling and Simulation (H. Tian, Editor),* pp. 113-117.

16 Kalko, E. K. V. and H. U. Schnitzler (1993). "Plasticity in echolocation signals of European pipistrelle bats in search flight: implications for habitat use and prey detection." Behavioral Ecology and Sociobiology **33**: 415-428.

17 King , E. G. and C. E. Rogers (1986). "In-house Report of the ARS National Heliothis program." U.S Dept of Agriculture: 39.

18 Kunz, T. (1982). Ecology of bats. New York, Plenum Press.

19 Kunz, T., J. Whitaker, et al. (1995). "Dietary energetics of the insectivorous Mexican free-tailed bat (Tadarida brasiliensis) during pregnancy and lactation." Oecologia **101**: 407-415.

20 Jorge Fernandez-Cornejo and Willaim. D. M. (2000). "Genetically Engineered Crops for Pest Management in U.S. Agriculture: Farm-Level Effects." U.S. Department of Agriculture. Agricultural Economic Report No. 786.

21 Lee, Y. and G. McCracken (2001). "Timing and Variation in the Emergence and Return of Mexican Free-tailed Bats, Tadarida brasiliensis mexicana." Zoological Studies **40**: 309-316.

22 Lee, Y. and G. McCracken (2002). "Foraging activity and food resource use of Brazilian free-tailed bats, Tadarida brasiliensis(Molossidae)." Ecoscience **9**: 306-13.

23 Lee, Y.-F. and G. F. McCracken (2005). "Dietary Variation of Brazilian free-tailed bats links to migratory populations of pest insects." Journal of Mammalogy **86**: 67-76.

24 M A Rankin, J. C. A. Burchsted(1992). "The Cost of Migration in Insects." Annual Review of Entomology **Vol. 37: 533-559**.

25 Marvier, M. (2001). "Ecology Of Transgenic Crops." America Scientist **89**: 160-167.

26 Neuweiler, G. (1990). "Auditory adaptations for prey capture in echolocating bats." Physiological Reviews **70**: 615-641.

27 Norberg, U., T. Kunz, et al. (1993). "The cost of hovering and forward flight in a nectar-feeding bat, Glossophaga soricina, estimated from aerodynamic theory." Journal of Experimental Biology **182**(1): 207-227.

28 Pennycuick, C. J. (1989). Bird flight performance : a practical calculation manual, Oxford University Press.

29 Raghavan, A. (2005). Modeling study of individual and group behavior of Brazilian free-tailed bats (Tadarida Brasiliensis) and dynamic

counting using real time infrared thermal video . Thesis -Electrical Engineering, University Of Tennessee ,Knoxville.

30 Robinson, M. F. (1990). "Prey selection by the brown long-eared bat (Plecotus auritus)." **28**: 5-18.

31 Roeder, K. D. (1962). "The behaviour of free flying moths in the presence of artificial ultrasonic pulses." Animal Behaviour **10**: 300-304.

32 Rosenfeld, A. and J. Pfaltz.(1996)."Sequential Operations in Digital Picture Processing." Journal Of the ACM **13**(4): 471-494.

33 Rydell, J., N. Skals, et al. (1997). "Hearing and bat defence in geometrid winter moths." Proceedings of the Royal Society B: Biological Sciences **254**: 83-88.

34 Thomas, S. (1975). "Metabolism during flight in two species of bats, Phyllostomus hastatus and Pteropus gouldii." The journal of experimental biology **63**: 273-293.

35 Thompson, M. J. A. (1982). "A common long-eared bat Plecotus auritus: moth predator-prey relationship." **107**: 87-97.

36 Schoener, T W . (1971). "Theory of feeding strategies." Ann rev ecol sys 2: 369-404.

37 Wahl, R. (1993). "Important Mexican Free tailed bat colonies in texas." Proceedings of 1989 national cave management symposium: 47-50.

38 Waters, D. A. (2003). "Bats and moths: what is there left to learn ?" Physiological Entomology **28**: 237-250.

39 Wolf, W., J. Westbrook, et al. (1990). "Recent Airborne Radar Observations of Migrant Pests in the United States." Philosophical Transactions of the Royal Society of London **328**: 619-630.

Vita

Haritha Kolli was born on 4th April 1984 in Hyderabad, India. She received her Bachelor of Technology degree in Electronics and Communication Engineering from the Jawaharlal Nehru Technological University in June 2005. She started working on her thesis under the guidance of her mentor Dr.Thomas G.Hallam at the Institute for Environmental Modeling Laboratory as a graduate assistant from August 2005. She enjoys programming in C, C++, VC++ and C#.Net. She graduated with a Master of Science Degree in Computer Engineering from University Of Tennessee in August 2007.