U.S.N.A --- Trident Scholar project report; no. 347 (2006)

The Search for Planet X: Testing Inferences from the Kuiper Cliff

by

Midshipman 1/c Eric A. Roe, Class of 2006
United States Naval Academy
Annapolis, Maryland

_____
(signature)

Certification of Adviser(s) Approval

Assistant Professor Jeffrey Larsen
Physics Department

_____
(signature)

_____
(date)

Professor Elise Albert
Physics Department

_____
(signature)

_____
(date)

Acceptance for the Trident Scholar Committee

Professor Joyce E. Shade
Deputy Director of Research & Scholarship

_____
(signature)

_____
(date)

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATE COVERED
	4 May 2006	

4. TITLE AND SUBTITLE
The search for Planet X : testing inferences from the Kuiper Cliff

5. FUNDING NUMBERS

6. AUTHOR(S)
Roe, Eric A. |q(Eric Andrew), 1983-

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)		8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSORING/MONITORING AGENCY REPORT NUMBER
US Naval Academy Annapolis, MD 21402		**Trident Scholar project report no. 347 (2006)**

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
This document has been approved for public release; its distribution is UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT: The goal of this project was to search a large archive of astronomical CCD imagery to test the validity of arguments regarding the existence of a hypothetical “Planet X.” As proposed by Brunini and Melita (2002), this object would be a low inclination Mars-sized body residing in the Kuiper belt between 55 and 75 astronomical units from the Sun. Resonances with this hypothetical planet form a convenient explanation of the “Edgeworth-Kuiper Cliff,” the observation that no low eccentricity Kuiper Belt objects are found more than 47 AU from the Sun despite numerous searches. Although the proposed “Planet X” is relatively bright, its distance means that it could easily have been mistaken for a stationary object by previous searches. This project utilized the image archives of the University of Arizona's Spacewatch Project, a near-Earth asteroid survey based at Kitt Peak National Observatory, including data from May of 2003 to April of 2006. The survey images were taken with a CCD mosaic camera on a 0.9 meter telescope with a 2.7 square degree field of view at 1” resolution to a limiting magnitude of 21.7. A unique feature of this survey was its reimaging of the same sky area every 3 to 7 nights. The survey’s rate sensitivity was 20-60 times greater than existing surveys. The final search covered approximately 10,600 square degrees of sky within 10 degrees of the ecliptic. An automated motion detection program was modified for this multinight search and processed approximately 3 terabytes of imagery. Nonetheless, visual validation of the candidates was still required. The efficiencies of the software and of the human reviewer were measured as functions of rate, field clutter and signal-to-noise. Quality control on the data was performed and methodologies were developed to aid in determining the unique coverage of the search and the probability of detecting the planetary candidate. One of the major results of this survey was the discovery of 2003 MW12, a significant Kuiper Belt asteroid, currently the tenth largest known in the Classical group. In addition, several known large Kuiper Belt objects and Centaur asteroids were detected. Followup observations of 2003 MW12 were obtained in February of 2006 at the Spacewatch 1.8 meter telescope on Kitt Peak.
.

14. SUBJECT TERMS: Astronomy, Edgeworth-Kuiper Belt, Kuiper Cliff, resonance, survey, Planet X	15. NUMBER OF PAGES **156**
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT

Abstract

The goal of this project was to search a large archive of astronomical CCD imagery to test the validity of arguments regarding the existence of a hypothetical "Planet X." As proposed by Brunini and Melita (2002), this object would be a low inclination Mars-sized body residing in the Kuiper belt between 55 and 75 astronomical units from the Sun. Resonances with this hypothetical planet form a convenient explanation of the "Edgeworth-Kuiper Cliff," the observation that no low eccentricity Kuiper Belt objects are found more than 47 AU from the Sun despite numerous searches. Although the proposed "Planet X" is relatively bright, its distance means that it could easily have been mistaken for a stationary object by previous searches.

This project utilized the image archives of the University of Arizona's Spacewatch Project, a near-Earth asteroid survey based at Kitt Peak National Observatory, including data from May of 2003 to April of 2006. The survey images were taken with a CCD mosaic camera on a 0.9 meter telescope with a 2.7 square degree field of view at 1" resolution to a limiting magnitude of 21.7. A unique feature of this survey was its re-imaging of the same sky area every 3 to 7 nights. The survey's rate sensitivity was 20-60 times greater than existing surveys.

The final search covered approximately 10,600 square degrees of sky within 10 degrees of the ecliptic. An automated motion detection program was modified for this multi-night search and processed approximately 3 terabytes of imagery. Nonetheless, visual validation of the candidates was still required. The efficiencies of the software and of the human reviewer were measured as functions of rate, field clutter and signal-to-noise. Quality control on the data was performed and methodologies were developed to aid in determining the unique coverage of the search and the probability of detecting the planetary candidate. One of the major results of this survey was the discovery of 2003 MW12, a significant Kuiper Belt asteroid, currently the tenth largest known in the Classical group. In addition, several known large Kuiper Belt objects and Centaur asteroids were detected. Followup observations of 2003 MW12 were obtained in February of 2006 at the Spacewatch 1.8 meter telescope on Kitt Peak.

Key Words: Astronomy, Edgeworth-Kuiper Belt, Kuiper Cliff, resonance, survey, Planet X

Contents

Acknowledgements

I would like to thank, first and foremost, my advisers, Assistant Professor Jeffrey Larsen and Professor Elise Albert for their time and effort in helping me complete this project. I would like to thank Prof. Larsen for pulling me from the raging waters of technical jargon and archaic computer code time after time after getting in over my head. I would like to thank Prof. Albert for agreeing to help with the paperwork for the project, including critiquing my torturous rough drafts for me and playing audience to more than one monotonous presentation practice.

I would also like to thank my parents for the support they have given me over the time I have spent at the Academy. Although I don't get to see them as often as I would wish, their dedicated telephone support more than makes up for it. I would especially like to thank my mother for devoting so much business trip free time to coming to visit.

Finally, I would like to thank all the members of the Trident Committee who have put effort into reviewing my project and making helpful suggestions, and the members of the Spacewatch Project, without whose help I would not have been able to undertake the project in the first place.

Chapter 1: Background

1. Search for a Tenth Planet: Testing Inferences from the Kuiper Cliff

This project was a search for a possible planet located in the Kuiper Belt, and aimed to set strong limits on three competing theories regarding the proposed planet's existence. In order to complete these goals, the project involved adapting existing software, originally designed to detect moving objects in a single night, to efficiently detect moving objects over multiple nights. This software was used to search approximately 10,600 square degrees of sky, within which the distant planet could lie, while simultaneously plotting coverage in position and phase space. Upon completion, analysis of sky coverage yielded the volume of space searched and provided new constraints on the existence of the hypothetical planet and the structure of the Kuiper Belt. To understand arguments surrounding the possible existence of a tenth planet, the following sections explore background information pertaining to solar system structure and dynamics.

2. History of the Kuiper Belt and Discovery of the Kuiper Cliff

The debate surrounding the existence of a tenth planet is a relatively recent development, for not even Pluto, dubbed 'Planet X' by Percival Lowell, was discovered until 1930. Twenty years later, in 1950, Dutch astronomer Jan Oort hypothesized the existence of a distant group of comet-like objects orbiting the Sun a great distance past Pluto. The Oort Cloud, postulated to exist between 50,000 and 100,000 AU from the Sun and contain millions of comets left over from the formation of the solar system, is named after Jan Oort in recognition. Oort's theories

also influenced the thinking of one of his contemporaries, Gerard Kuiper, who proposed the existence of a closer group of comets, with different orbital characteristics than Oort's group, orbiting just beyond Neptune. Agreeing with Kuiper's beliefs that this group represents remains from the formation of the solar system, astronomers in the early 1980's ran computer simulations of the formation of the solar system. Their models supported Kuiper's theories, in that they consistently predicted the formation of an icy group of debris left over from the Sun's accretion disk which would not have coalesced into planets at the fringes of the solar system. Despite many searches, it was not until 1992 that the first Edgeworth-Kuiper Belt object (EKBO), a 150 mile long body named (15760) 1992 QB1 was discovered by astronomers David Jewitt and Jane Luu.[1] Since then, approximately 500 EKBOs of an estimated population of 70,000 bodies with diameter greater than 100 kilometers have been discovered, as shown in the recently generated model of the outer solar system depicted in Figure 1.

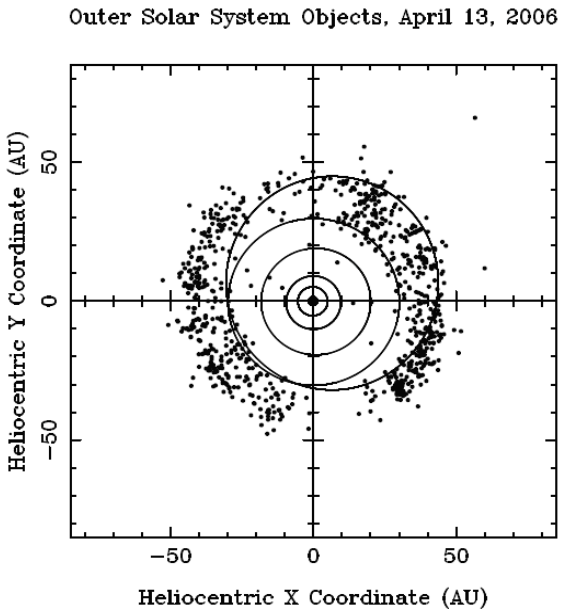

Figure 1: This image represents the observed structure of the Kuiper Belt, and is shown to scale with the planets orbiting in the outer solar system. Note that, by showing the outer solar system to scale, the depiction of the inner solar system (Mercury, Venus, Earth, Mars, and the asteroid belt) becomes too small to be represented. Data to produce this plot was gathered from the International Astronomical Union (IAU) Minor Planet Center.

Some of these objects are relatively large (such as Sedna and Quaoar), and are shown to scale with the Moon, Earth and Pluto in Figure 2. Indeed, an object known to be larger than Pluto, 2003 UB_{313}, was discovered in this region. These objects, due to their size, continue to fuel the debate surrounding the currently ambiguous definition of a planet and suggest that Pluto, instead of being a distinct planet, is merely another Kuiper Belt object.

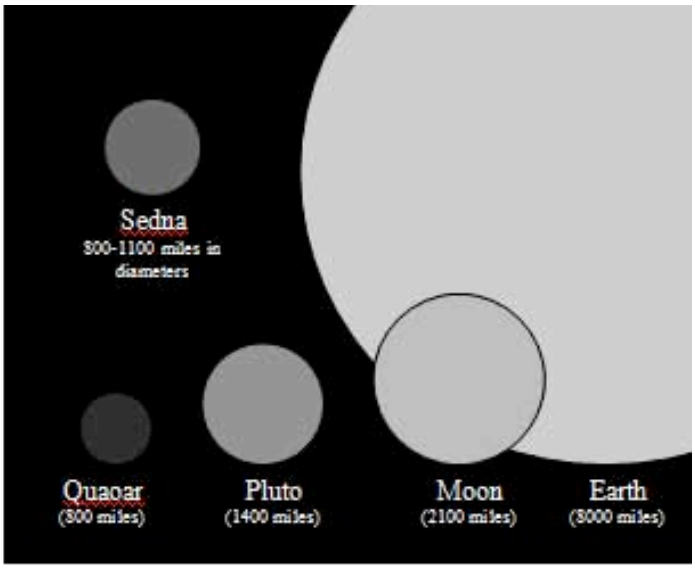

Figure 2: Sedna is nearly the same size as Pluto, both of which are smaller than the Moon. Quaoar is not much smaller than Sedna. The relative sizes of these objects illustrate the debate surrounding the definition of a planet, brought about by Pluto's recently determined diminutive size and further emphasized by the discovery of nearly Pluto-sized bodies in its general vicinity. Adapted from http://www.jpl.nasa.gov/releases/2004/85.cfm.

Analyzing the characteristics of known EKBOs in 1998, David Jewitt, Jane Luu, and Chadwick Trujillo defined three distinct classifications of EKBOs: the plutinos, trapped in a 2:3 mean-motion resonance with Neptune, scattered EKBOs (SEKBOs), highly eccentric (elliptically orbiting) objects with semi-major axes of approximately 90 AU and perihelia centered at 35 AU, and classical EKBOs (CEKBOs), a group of low to moderately eccentric orbiting objects with semi-major axes between 41 and 46 AU that comprise nearly two thirds of the known EKBO

distribution.[2] Of these three groups, the CEKBOs are of particular note. When observing their orbital parameters, Trujillo, Luu, and Jewitt noticed that, contrary to computer simulations, the number of CEKBOs discovered at distances greater than 50 AU abruptly fell to zero. In their 1998 paper entitled "Large Kuiper Belt Objects: The Mauna Kea 8K CCD Survey," they concluded that either there was a sharp truncation of the Kuiper Belt at 50 AU, or that the size of the CEKBOs decreased rapidly beyond 50 AU, making them harder to detect.[3] Surveys undertaken since then, which would have been able to detect the smaller CEKBOs they predicted, have produced the same results.[4] This apparent gap in the Kuiper Belt, dubbed the Kuiper Cliff for the way in which the plot of eccentricity versus semi-major axis of observed CEKBOs (shown in Figure 3) dramatically reduces to zero, led Trujillo, Luu, and Jewitt to the same conclusions in their second study in 2001.

[2]Trujillo, C.A., D.C. Jewitt and J.X. Luu, "Properties of the Trans-Neptunian belt: Statistics of the CFHT survey," in *Astronomical Journal*, Vol. 122 (2001): pp. 157-473.
[3]Jewitt, David, Jane Luu and Chadwick Trujillo, "Large Kuiper Belt Objects: The Mauna Kea 8K CCD Survey," in *Astronomical Journal*, Vol. 115 (1998): pp. 2125-2135.
[4] Brunini, Adrian and M.D. Melita. "The Existence of a Planet beyond 50 AU and the Orbital Distribution of the Classical Edgeworth-Kuiper-Belt Objects." In Icarus, Vol. 160 (2002): pp. 32-43.

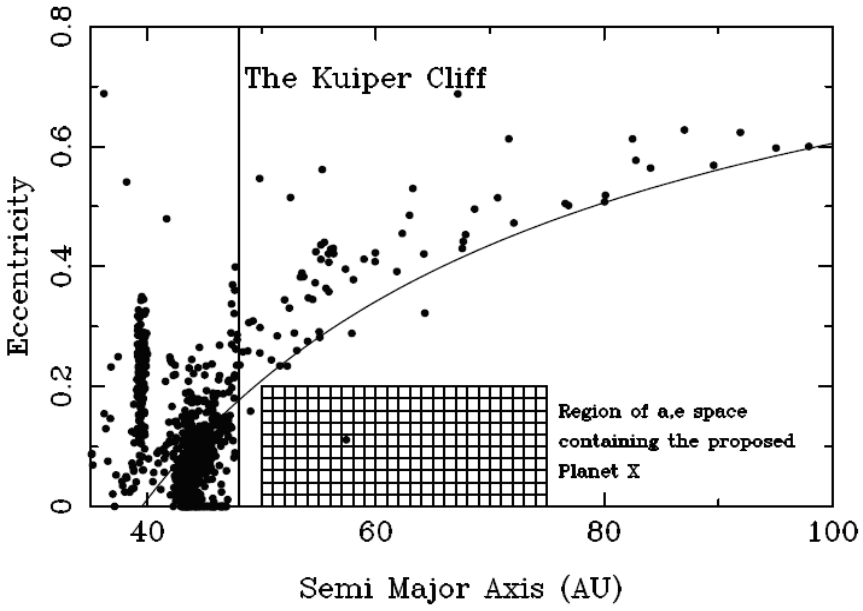

Figure 3: Observed Kuiper Belt Objects, plotted as eccentricity v. semi-major axis, in which the separate groups of EKBO's are labeled. The Kuiper Cliff is prominent, as the number of CEKBOs reduces to virtually zero past 47 AU. While highly eccentric EKBOs still exist past this distance, Brunini and Melita conclude that the sharp decline can only be the result of gravitational interaction with a massive body such as Planet X. From Brunini, Adrian and M.D. Melita. "The Existence of a Planet beyond 50 AU and the Orbital Distribution of the Classical Edgeworth-Kuiper-Belt Objects." The sole object in the Planet-X (a,e) region is a high eccentricity object, 2004 XR_{190}.[5]

In this paper, they again suggested several plausible scenarios which could have been responsible for creating such a gap, but ultimately concluded that the Kuiper Belt has an edge at 50 AU and that the gap represents a physical absence of any CEKBOs. Facing this conclusion, the team attempted to determine a process which could create such a gap.

First, they considered the possible role of resonances caused by Neptune. Given the 3:2 and 2:1 resonances between Neptune and groups of EKBOs already found in the Kuiper Belt, Trujillo et al. concluded that these resonances could only have affected portions of the Kuiper Belt interior to these objects, as these EKBOs would have migrated outwards to their current distance over the age of the solar system, just as the planets have done. As a result, these resonances could not

[5] Allen, R.L. et al, "Discover of a Low-Eccentricity, High-Inclination Kuiper Belt Object at 48 AU," in *The Astrophysical Journal*, Vol. 640 (2006): pp. L83-L86.

have served to create the observed Kuiper Cliff because it lies outside both the 2:1 and 3:2 resonances and would have migrated with them. With this explanation for the formation of the Kuiper Cliff discredited, Trujillo, Jewitt, and Luu also rejected the notion that this truncation was the "effect of a close stellar encounter on the Kuiper Belt, suggesting that KBO orbits beyond 0.25 to 0.3 times the stellar perihelion distance would be disrupted and ejected [from the Kuiper Belt] for a variety of encounter inclinations."[6]

Trujillo, Jewitt and Luu's arguments were disputed by Adrian Brunini and M.D. Melita, who proposed in 2002 that the Kuiper Cliff was created by the gravitational influences of an undiscovered planetary-sized body orbiting in the region, noting that previously theorized close stellar encounters would not have created the sharply truncated Kuiper Cliff observed.[7] In their paper, Brunini and Melita suggest that the Kuiper Cliff was formed through resonances with the undetected planet, and argue that such a body could easily have eluded detection due to its slow sky motion. Speaking of their proposed planet, Brunini and Melita concluded that "if such an object is eventually observed—together with the EKBOs, centaurs (objects orbiting between Jupiter and Neptune), and comets—it would give important clues for the study of the outer Solar System."[8]

As a matter of fact, recent observations of the Kuiper Belt have resulted in the detection of several massive bodies with distant, highly inclined orbits. The largest of these new objects, temporarily designated 2003 UB_{313}, has been confirmed to be larger than Pluto, and was discovered by Chadwick Trujillo, David Rabinowitz, and Michael Brown in January 2005 and

[6] Trujillo, C.A., D.C. Jewitt and J.X. Luu, "Properties of the trans-neptunian belt: Statistics of the CFHT survey," in *Astronomical Journal*, Vol. 122 (2001): pp. 157-473.
[7] Brunini, Adrian and M.D. Melita, pp. 32-43.
[8] Brunini, Adrian and M.D. Melita, p. 40.

announced in July of that year.[9] Having a highly irregular orbit that is inclined 44 degrees above the plane of the solar system and a semi-major axis that ranges from 38 AU (the approximate distance of Pluto) to 97 AU (where it is currently located), its discovery supports theories arguing the existence of many undetected large bodies in the Kuiper Belt. Additionally, two other massive bodies, 2005 FY9 and 2003 EL61, have been discovered and were announced shortly following the announcement of UB_{313}. These bodies, although smaller (at roughly three quarters and one quarter the size of Pluto respectively) have orbital inclinations (28 and 29 degrees) and semi-major axes (between 39 and 52 AU, and 35 and 52 AU) more similar to scattered disk objects, thus leaving room for Brunini and Melita's proposed Planet X. Indeed, speaking of the recent discoveries in an interview for the popular astronomy magazine *Sky and Telescope*, renowned Kuiper Belt expert S. Alan Stern commented "I expect things at least as large as Earth to be in distant orbits"[10] around the Sun. With so much support, a closer look into the gravitational resonances argued by Brunini and Melita to be the cause for the formation of the Kuiper Cliff, is warranted.

3. Dynamic Interaction: Gravitational Perturbation and Resonance

The resonances hypothesized by Brunini and Melita are the result of complex gravitational interactions within the multi-body celestial system found in the Kuiper Belt, and require a basic understanding of the fundamental orbital properties caused by gravitational interaction. Gravitational perturbation, the method through which resonance is created, stems from the gravitational force one massive object exerts on another, and serves to alter the orbits of

[9] Brown, M.E., C.A. Trujillo, and D. L. Rabinowitz, "Discovery of a Planetary-Sized Object in the Scattered Kuiper Belt," in *The Astrophysical Journal*, Vol. 635 (2005): pp. L97-L100.
[10] Tytell, David, "The New Kings of the Kuiper Belt," in *Sky and Telescope*, October 2005, p. 31.

both objects over time. This force, which is proportional to the inverse square of the distance between the two objects, is strongest at conjunction (when two orbiting objects are at their point of closest approach) and weakest at opposition (point of greatest distance). Gravitational perturbations act to alter each body's orbital properties in a way that evens out the periods of strong and weak interaction to regular intervals. With conjunction occurring at regular intervals, the point at which an object feels the strongest gravitational force from its neighbors occurs at the same place every orbital period. This leads to a system of driven harmonic oscillatory motion, with repeated forces at congruent locations and regular intervals of time eventually moving objects towards relative points of stability where the forces are more balanced against each other. If, however, conjunction does not occur at regular intervals, the point at which they feel the strongest perturbation will migrate along each object's orbit, with the effect cancelling out over time. If this is the case, C.D. Murray's and S.F. Dermott's *Solar System Dynamics* textbook explains that the tangential component of the gravitational force experienced by a body before conjunction will not equal the tangential component of the force experienced by that body after conjunction.[11] As a result, Murray and Dermott conclude that "the net result of the encounter is an increase in angular momentum of the . . . object and a decrease in its mean angular velocity."[12] Due to the net tangential force remaining, the perturbed particle's angular momentum (L) and mean orbital velocity (n) will be transferred from one body to another, changing the orbital properties of both.

By equalizing the intervals between conjunction and opposition, perturbation alters the orbits of objects in a system to bring about a state of stability, represented by a resonance.

[11]Murray, C.D. and S.F. Dermott, *Solar System Dynamics*, Cambridge University Press: United States, 1999, p. 327
[12]Murray, C.D. and S.F. Dermott, p. 332

Resonance is an integer ratio between any two periods of motion, commonly seen in the solar system as a ratio between two periods of rotation or revolution. A prominent example of resonance is shown in Figure 5, which illustrates the Moon's 1:1 spin resonance with the Earth.

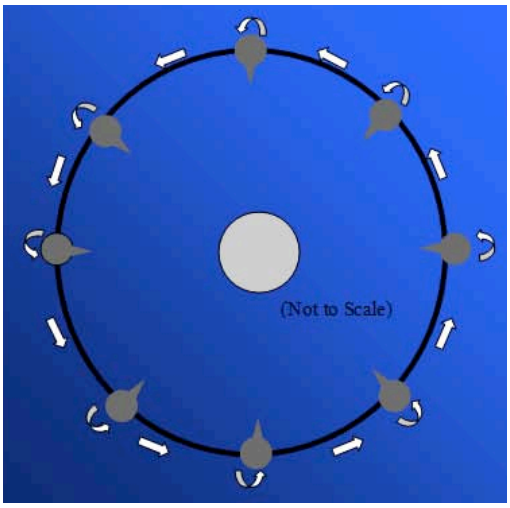

Figure 4: This picture shows the Moon's 1:1 spin resonance with the Earth. For every time the Moon rotates once about its axis of rotation, it also completes one orbital revolution about the Earth. As shown by the mountain drawn on one face of the Moon, the 1:1 spin resonance results in a synchronous orbit with the same face pointing towards Earth at all times. This is why we can only see one 'side' of the Moon.

Because resonance represents a state of stability, a system not in resonance will migrate over time through the gravitational perturbations amongst the bodies in that system. It will eventually reach a resonant state, just as the Moon has migrated into a resonant state with the Earth. Another prominent example of resonance in the solar system can be found in the main asteroid belt, located between Mars and Jupiter. In this belt, which has been studied extensively, different resonances push the asteroids to points of stability, forming concentrated groups at specific distances from the Sun. The less massive asteroids have been perturbed away from resonant states with Jupiter over time, and have formed the Kirkwood Gaps shown in Figure 5.

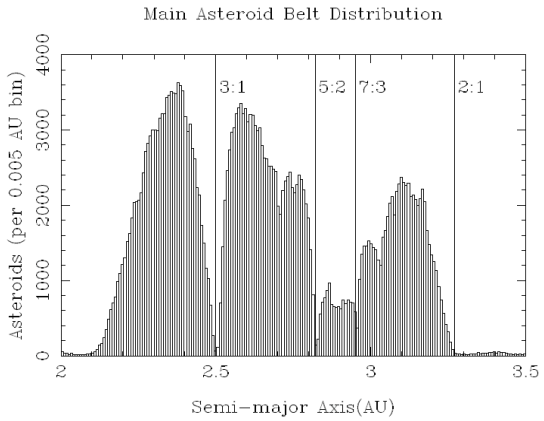

Figure 5: This Figure is a plot of the distribution of the Main Asteroid Belt in semi-major axis versus quantity of asteroids. The Kirkwood Gaps, caused by resonances with Jupiter that are listed on the plot, are readily apparent as sharp declines in the quantity of Main Belt asteroids at semi-major axes of 2.5, 2.81, 2.95, and 3.8 AU. Information from http://cfa-www.harvard.edu/iau/MPCORB.html.

4. The Case for 'Planet X:' The Kuiper Belt, Brunini and Melita

The Kuiper Cliff was identified from studies of EKBO orbital properties since the first Kuiper Belt object was discovered in 1992. The structure of the Kuiper Belt is revealed by plotting each object's orbital eccentricity versus its average distance from the Sun, as shown in Figure 3. In this plot, it is interesting to note a seemingly precipitous decrease in low eccentricity CEKBOs at approximately 47 AU from the Sun. The gap created by the virtual absence of Kuiper Belt Objects has been christened the 'Kuiper Cliff.' Analyzing the Kuiper Cliff, Trujillo, Jewitt, and Luu, M.D. Melita and A. Brunini recognized that such a physical feature could be the result of resonant sweeping caused by interaction between CEKBOs and a distant planet-sized body, just as resonances within the main asteroid belt stem from their

interaction with Jupiter and are responsible for the sharp, truncated, Kirkwood Gaps observed there (shown in Figure 5).[13]

While Trujillo, Jewitt, and Luu had proceeded on the premise that the Kuiper Cliff could have formed from a close stellar encounter, Brunini and Melita attempted to determine a viable candidate for a distant planet whose gravitational influence could recreate the observed structure of the Kuiper Cliff. In their 2002 study, *The Existence of a Planet Beyond 50 AU and the Orbital Distribution of the Classical Edgeworth-Kuiper Belt Objects*, Brunini and Melita utilized a computer modeling program based on a symplectic integrator. The symplectric integrator produces differential equations of motion by using the Hamiltonian to describe the interaction of the energies in the system, and not the interaction of gravitational forces. This method was first developed by John Chambers in his paper, *A Hybrid Symplectic Integrator that Permits Close Encounters between Massive Bodies* (1998). Using Chambers' symplectic integrator as a basis for their computer model, Brunini and Melita ran computer simulations of an encounter between a massive body and a group of CEKBO test particles of semi-major axes between 35 and 90 AU.

In all, Brunini and Melita ran four computer simulations over a time span of one billion years, varying the mass, semi-major axis, eccentricity, and inclination for the hypothetical object each run. After only their fourth attempt, they were able to match the observed data to a very precise degree, accurately producing the Kuiper Cliff from Figure 3 in Figure 6 below. Their results, using a planet 10% as massive as Earth (the approximate mass of Mars), with a migrating semi-major axis between 55 and 75 AU, an eccentricity (ε) of 0.2, and an inclination of 10 degrees from the ecliptic (the orbital plane of the solar system), are shown in Figure 6.

---

[13] Brunini, Adrian and M.D. Melita. "The Existence of a Planet beyond 50 AU and the Orbital Distribution of the Classical Edgeworth-Kuiper-Belt Objects." In *Icarus*, Vol. 160 (2002): pp. 32-43.

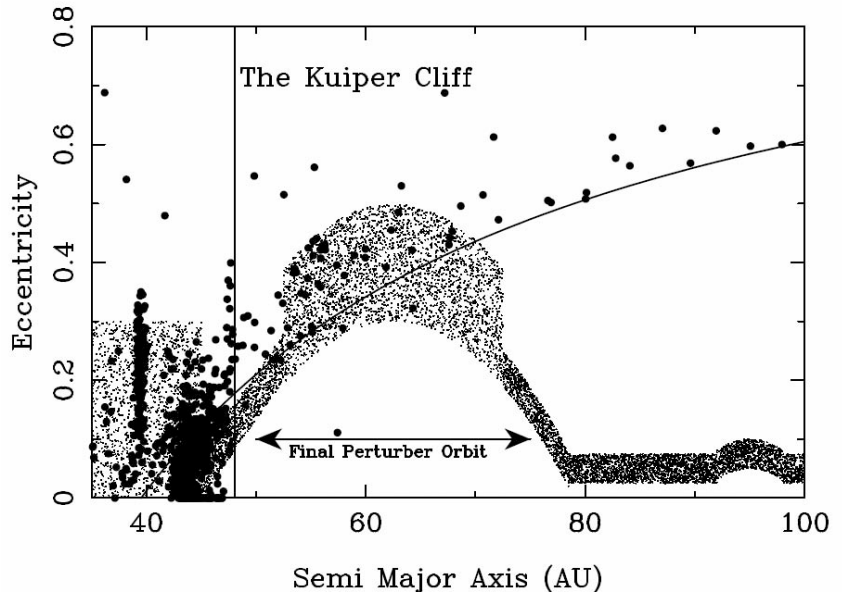

Figure 6: Computer simulation of a body of $M = 0.1 M_{sun}$, $e = 0.2$, a=55-75 AU, and $i = 10°$. The open circles represent the simulated particles, while the solid circles represent observed KBO's. The pixelated region is the area bounded by Brunini and Melita's simulated objects, and represents the region where these objects could be found. Note that Brunini and Melita were able to accurately recreate the Kuiper Cliff region which is void of any CEKBOs. Adapted from Brunini, Adrian and M.D. Melita, "The Existence of a Planet beyond 50 AU and the Orbital Distribution of the Classical Edgeworth-Kuiper-Belt Objects."

The only difference between the observed and simulated data is represented by reemergence of low eccentricity CEKBOs at distances greater than 70 AU. This difference is entirely plausible, however, as any CEKBO orbiting at such a distance would be so faint and slow that it could easily not have been detected by any observational study conducted to date or so rare as to be seen only by systematically surveying large quantities of the sky.

5. Competing Theories: Explanations for the Location of Planet X

After defining the required properties of Planet X, Brunini and Melita discussed why, if a significantly massive body did act to perturb the Kuiper Belt objects to create the structure

observed today, it has not been located. The limiting magnitude of the latest searches for Kuiper Belt objects should have allowed them to detect an object as large as Mars, orbiting between 55 and 75 AU, if they are able to detect faint comets out to 80 AU. However, those searches did not have high enough sky coverage to make a conclusive argument regarding such an object's existence. In any case, current understanding of the formation of our solar system holds that the formation of a terrestrial (rocky) body the size of Planet X would be impossible in its proposed location. In this case, Planet X would have formed closer to the Sun, and then migrated outwards to its current location over the lifetime of the solar system. The transportation process occurs through the transfer of angular momentum, made possible through gravitational perturbations.

Additionally, Brunini and Melita posited that gravitational drag could also have led to the migration of Planet X. They argue that as Planet X passes through the Kuiper Belt, it experiences a net gravitational drag force from the combined mass of the CEKBOs. This drag force acts to slow the mean orbital velocity of the perturbing body, just as described by Murray and Dermott, thus altering its orbit to a slightly more eccentric one. Uranus and Neptune would also transfer some angular momentum to the body, further altering the orbit, and ultimately causing the body to migrate outwards from its original orbit. Since this process repeats with every orbital period, it represents a method by which celestial mechanics allows a planet to migrate through the solar system. This is not a revolutionary idea, for it has already been shown that the nine known planets have migrated from their original orbits over the age of the solar system to their current location. Conducting computer simulations of their transportation mechanism, Brunini and Melita produced results that supported their hypothesis and further strengthened their case for Planet X.

More recently, an alternative formation theory for Planet X at its current location has been proposed by S. Alan Stern in his paper, *Regarding the Accretion of 2003 VB (Sedna) and Like Bodies in Distant Heliocentric Orbits*, published in January 2005.[14] In his paper, Stern concludes that, contrary to previous works, "Sedna and even larger bodies could indeed have grown from kilometer-class planetesimal seeds out to distances as great as 500 AU."[15] If Sedna, a CEKBO comparable in size to Pluto, could form from accretion, it is also plausible that Planet X could have accreted as well. Stern also comes to this conclusion by hypothesizing that Sedna could be one among many larger sized bodies orbiting in the Kuiper Belt region. As a result, conducting a survey for these objects stands as one of the major goals of this project and will be the only way to determine the validity of these competing theories and to discriminate between them.

[14] Stern, S. Alan, "Regarding the Accretion of 2003 VB (Sedna) and Like Bodies in Distant Heliocentric Orbits," in *The Astronomical Journal*, Vol. 129 (2005): p. 526-529.
[15] Stern, S. Alan, p. 529.

Chapter 2: Collection of Data and Revision of SLOSUR

1. Collection and Analysis of Data: SLOSUR

Although Brunini, Melita and Stern have proposed credible arguments for the existence of a tenth planet that are backed by observations and theoretical calculations, there is no guarantee that the planet is still located in the solar system. Even Brunini and Melita's transportation mechanism implies that, because planets migrate over time, Planet X could easily have passed through the Kuiper Belt some time in the past, leaving its mark on the Kuiper Belt objects there, and continued out of the solar system and into interstellar space.[16] On the other hand, no observational programs have yet covered the expanse of sky necessary to refute or support Brunini and Melita's claims. To cover the region of sky proposed in Brunini and Melita's paper, this survey's approach was to search of the entire sky within 10 degrees of the ecliptic, covering between 10000 and 20000 square degrees. This was accomplished by using the Spacewatch project's partitioning of the sky into smaller segments, called regions. Examples of this are shown in Figure 7, which shows the areas of individual images taken and then repeatedly imaged over successive weeks to distinguish the proper motion of any object within the region. The data for this survey was collected at Kitt Peak National Observatory in Arizona, from March 2003 to April 2006, as part of the normal operations of the Spacewatch Project. The Spacewatch project is an ongoing survey for asteroids and near-Earth objects conducted by the Lunar and Planetary Laboratory of the University of Arizona in Tucson, which maintains both a 0.9 meter and a 1.8 meter telescope on nearby Kitt Peak for this purpose.

[16] Brunini, Adrian and Julio A. Fernandez. "Perturbations on an Extended Kuiper Disk Caused by Passing Stars and Giant Molecular Clouds."

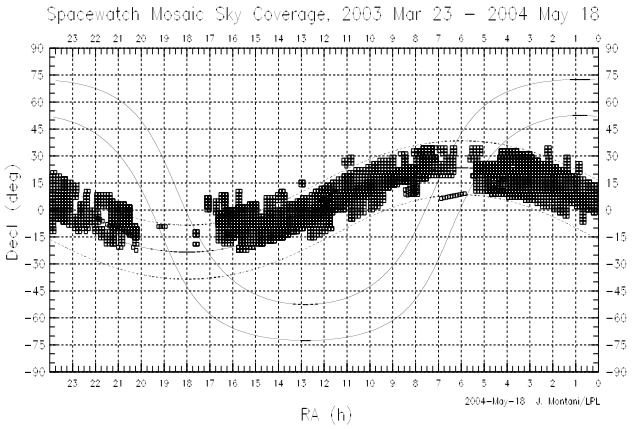

Figure 7: This plot represents approximately one year's worth of imaging coverage taken by the Spacewatch group in Kitt Peak, Arizona, shown from north to south and east to west. A total of two years of data was analyzed over the course of this project. Image from J. Montani of the Spacewatch project.

Survey data was collected using the 0.9 meter telescope, which records images of the sky with four E2V Technologies CCD42-90-I-941 charge-coupled-device (CCD) cameras mounted in the focal plane of the telescope. The operation of these CCD cameras is similar to a digital camera, but they have much greater light sensitivity due to decreased read noise and thermal cameras. Incoming photons striking the silicon sensor chip of the CCD create an electrical charge which is temporarily stored at that location. At the end of the collection run, the charge is moved off of the sensor and read into electronics which converts the amount of charge on each pixel into a measurement of the intensity of light at that location and a computer renders an image. The telescope has a plate scale of 1 arcsecond per pixel, and a total field of view of approximately 2.7 square degrees. This allows the 0.9 meter telescope to cover an area of the sky approximately ten times the size of the moon in a single exposure, yielding a rate of collection nine times faster than many wide field imaging telescopes. The four CCD chip layout in the optical plane of the 0.9 meter telescope is shown in Figure 8.

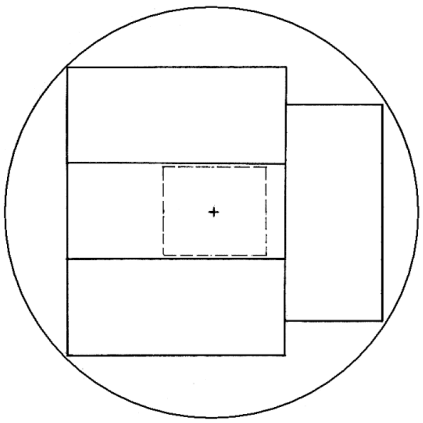

Figure 8: A field of view comparing the area covered by a telescope using one CCD camera, to the .9 meter telescope used by the Spacewatch Project, which uses four CCD cameras to cover an area nine times larger. The original CCD is placed in the region, while the four boxes surrounding it are representations of the four CCD cameras mounted on the .9 meter telescope. Obtained from http://spacewatch.lpl.arizona.edu/scopes.html.

The larger area imaged by this combination of four CCD cameras reduced the number of images that needed to be taken to cover the amount of sky for Planet X. With the planet thought to reside within 10 degrees of the ecliptic, it could inhabit a region of 10000-20000 square degrees of sky. Using a conventional telescope, with a field of view of approximately 0.4 square degrees, it would take nearly 25,000 images to cover this area. However, the Spacewatch 0.9 meter telescope can cover the same sky area in approximately 3700 images. For this reason, the 0.9 meter telescope was well suited to this search. For this project, six images of the same 2.7 square degree area of sky were taken on two nights, with three images per night, as part of the normal survey plan. The time period between nights, called the 'cadence,' was crucial for this project, as it allowed for the detection of slow object motions in the images and ultimately determined the search rate sensitivity. With a cadence from 3 days to one week and a faint limiting magnitude, the telescope is rate sensitive to Mars sized objects out to a distance of 130 AU, and Jupiter sized planets to a distance of 1600 AU (with the difference being the differing amount of sunlight reflected between the two). By using a cadence based on days rather than

hours, the survey was not able to detect typical main belt asteroids, which move at a sky rate of 0.2 degrees per day (approximately half the size of the full moon per day). Instead, it was ideal for Planet X like objects, moving at a rate of .015 degrees per day. After being gathered from the telescope facilities at Kitt Peak, the images were transported to Annapolis for processing.

Each 81 MB image consists of 17 million pixels and was stored on a 400 GB removable hard drive for transport. The 400 GB hard drives were loaded onto computers in a laboratory in the basement of Chauvenet Hall at the Naval Academy. This lab consisted of six server computers running Fedora Core 4 Linux (Kernel Version 2.6.11-1.1369_FC4), each equipped with an Intel Model 630 Pentium 4 3.0 GHz EM64T Processor, 2 GB of DDR2 RAM, a 150 GB SATA system hard drive, and three 400 GB removable hard drive bays, yielding a net storage capacity of 1.2 TB per computer. All six computers were cross-connected on an isolated 1000 Base-T network, which allowed each computer to see the data on the other five, making it possible to use all six computers to process data stored on any of the hard drives. With the combined computing power of all six computers and online storage of all imagery, the time required to process and analyze the data was greatly reduced.

Two mosaic software programs, named MOSAF (Mosaic Astrometry Finder) and SLOSUR (Slow Survey), were used to analyze the raw imagery. MOSAF created a catalog of all the stars on the six time-lapsed images, and calculated the precise position of the image on the sky by identifying stars of known position in the image, called astrometry stars. MOSAF did this as part of the real-time data acquisition pipeline at Kitt Peak, and would raster scan each image, examining the charge at each pixel and locating adjacent pixel regions representing stars, galaxies, and other celestial objects. SLOSUR is a C based program run on the Linux

workstations as a post-processing step. It took the two nights of MOSAF data, identified moving candidates, and created a list of potential Planet X-like objects. SLOSUR is the Planet X detection program modified from the regular asteroid detection program, MOSSUR (Moasic Survey), used by Spacewatch in real time. SLOSUR took two nights of MOSAF catalogs and created a list of all objects which appeared to be effectively stationary (moved less than 3 arcseconds in 2 hours) for each night. It then compared the two catalogs of stationary objects, searching for objects which (while stationary on one night) could not be found on the other night in the same position. These unmatched "matched objects" were placed in two lists, one for each night. the motion detection of a position in each night, whose net motion would be consistent with that of Planet X. If the orbit determined by Bernstein and Khushalani (2000) seemed appropriate, the candidate was kept. Candidate motions were placed in a review file which was an archive file containing small "postage stamp" images of the candidates from each image from each position on each night as well as all the image catalog information. Finally, these review files were made available for manual validation by a reviewer.

2. Revising SLOSUR: The Application of Filters

The first immediate goal of the project was to revise the SLOSUR program to increase its efficiency and decrease the number of candidates which had to be reviewed individually. Prior to the beginning of the survey, the SLOSUR program was operable but inefficient. Taking approximately 24 hours to process a single region (six passes over two nights), it would generate potential candidate review lists that numbered in the thousands for each region processed. Visually inspecting thousands of candidates from each region was not viable. To solve this

problem, SLOSUR needed to more successfully recognize objects that were not potential candidates. This was achieved by rewriting the matching process and applying several new candidate filters.

The first revision involved changing the matching process SLOSUR used to identify moving objects in each image. Originally, SLOSUR did not require all three passes to match, but instead, objects in any two of the three images could be considered matches and listed as moving objects. This method was changed to considering only those objects whose locations remained the same over all three passes, hence removing any object at the detection limit not seen all three times (an effective strengthening of the required signal to noise ratio). The application of this change was motivated after initial image analysis with the original SLOSUR program revealed many transient artifacts that were falsely identified as potential candidates because they accidentally matched in two of the passes. These artifacts were mainly defects on the CCD image, such as pixel bleeding caused when a sufficiently intense object, such as a bright star, caused enough photons to be incident on the CCD sensor chip that the pixel at that location was saturated and some charge 'overflowed,' or bled into surrounding pixels. This effect is depicted in Figure 9 below.

Figure 9: These three images represent one object from each of the three passes from one night. The bright star on the left side of the frame has saturated those pixels on the CCD camera, and caused photons to bleed into pixels in the surrounding area, and along a row from left to right.

With the object changing slightly in every image, SLOSUR could frequently find a saturated object in two, but not all three passes. These luminous objects, then, were frequently labeled as potential candidates, and were tagged for visual inspection. With the added requirement that an object be matched in all three passes, SLOSUR ignored these transient effects.

Three filters were added to the SLOSUR program to reduce the number of false identifications: a prograde motion filter, a signal to noise ratio filter, and the application of range circles to pair two moving objects into a potential candidate. The prograde motion filter rejected any identified objects with prograde motion without having to compute an orbital calculation. As a consequence of the conservation of angular momentum from the formation of the solar system, every massive body in the solar system is revolving around the Sun in the counter-clockwise direction, as seen when looking down on the solar system from above. An object revolving in this direction is said to have prograde motion, while movement in the opposite direction is called retrograde motion. Any potential Planet X will most certainly move prograde around the Sun, but will look different due to the Earth's own higher velocity prograde motion. The speed and direction with which a potential candidate should move, as viewed from Earth, can be calculated using simple physics. Earth takes 365.25 days to revolve around the Sun and, as such, moves with a mean orbital velocity of 0.9865 degrees per day. Using Newton's second law and Newton's law of gravitation (Eq. 1) to derive the equation for orbital velocity (Eq. 2), the mean orbital velocity for a potential Planet X was calculated to be anywhere from 0.002 degrees per day at 55 AU from the Sun, to 0.0015 degrees per day at 75 AU from the Sun in the prograde direction, assuming circular orbits.

$$F = \frac{GMm}{r^2} = ma_c \qquad \textbf{Eq. 1}$$

Using $a_c = \frac{v^2}{r}$, where F is force, G is the gravitational constant, M and m are the masses involved, v is the velocity of the orbiting object, r is the radius of rotation, and a_c is the centripetal acceleration,

$$v_{orb} = \sqrt{\frac{GM}{r}} \qquad \textbf{Eq. 2}$$

To accommodate general orbits which may be non-circular, the Vis-Viva equation (shown as Eq. 3) is derived from the conservation of energy, where the total energy of the orbiting object is equal to the kinetic energy of the object minus the gravitational potential energy of the object:

$$E = \frac{1}{2}mv^2 - \frac{GMm}{r} = -\frac{GMm}{2a} \qquad \textbf{Eq. 3}$$

where a is the semi-major axis of the orbit and r is the distance to the orbiting object at a specific point in time. By rearranging the terms, the Vis-Viva equation becomes:

$$v = \sqrt{GM\left(\frac{2}{r} - \frac{1}{a}\right)} \qquad \textbf{Eq. 4}$$

Using the Vis-Viva equation, the orbital velocity for Planet X becomes anywhere from 0.00121 degrees per day, at a distance of 75 AU with a semi-major axis of 55 AU, to 0.002718 degrees per day, at a distance of 55 AU with a semi-major axis of 75 AU. Because its orbital velocity is smaller than that of the Earth, Planet X could never appear to be moving faster than the Earth, a situation which would cause the Planet to no longer be gravitationally bound to the solar system. As a result, any object with prograde motion was dismissed from this survey without fear of missing real objects.

A second filter, concerned with the signal to noise ratio of detected objects, was also applied in order to make SLOSUR more efficient. The ratio of signal to noise is commonly used to set a baseline in studies which serves to differentiate between valid and invalid data. In

this study, the signal to noise ratio was defined by Equation 5 as the summation of intensity (I) of an object over its pixels divided by that number of pixels (n) multiplied by the mean detection noise (σ) determined from a blank image of the sky.

$$S/N = \frac{\sum_n I_i}{n\sigma_i} \qquad \textbf{Eq. 5}$$

Both of these values were inherent to the CCD image, as the CCD digitally stored the intensity of charge read at every pixel in the image, and was easy to work with as a result. To set the limits of the signal to noise ratio for this study, the minimum number of incoming photons from Planet X must be estimated. This was found by considering the likely apparent magnitude of the planet. Apparent magnitude is determined largely by the albedo, or percentage of incident light that is reflected from the planet's surface, the planet's size, and its distance from the Sun. For Planet X, the attenuation of light traveling over the 55 to 75 AU path from the Sun to the planet, and then over the 54 to 76 AU path back from the planet to the Earth would decrease in intensity approximately as $\frac{1}{d^4}$, where d is the distance from the Sun, due to the spherical spreading of the wavefront as it propagates. Estimations were made of the size and albedo of Planet X. First, the diameter of Planet X was approximated to equal that of Mars, due to the equivalent masses of the two objects, and the fact that their densities would be similar, being rocky bodies of similar composition. Secondly, the albedo of Planet X was estimated to be no less than that of an asteroid, also due to its similar rocky composition. Any surface features present, such as ice caps or an atmosphere, would only serve to increase this albedo and make the planet brighter. With these estimations, Brunini and Melita calculated the minimum apparent visual magnitude for Planet X to be between 18 and 21 magnitudes.

Because the limiting magnitude of the 0.9 meter telescope was 21.7 magnitudes, there were objects detected which were fainter than any potential candidate could be. These objects were often identified as moving objects because small variations in their brightness between nights (caused in large part by external atmospheric conditions) served to move them in and out of the telescope's detection envelope. As a result, a signal to noise ratio which rejects objects fainter than 21st magnitude would improve the efficiency of the SLOSUR program without hindering the detection of a potential Planet X candidate. Since objects with an apparent magnitude of 21 have a signal to noise ratio of 5.5; any object detected which has a signal to noise ratio of less than 5.5 can be rejected as an object too faint for consideration. In applying this signal to noise ratio filter, it was possible that SLOSUR would reject some real objects, but they would have to be smaller than a Mars-sized Planet X candidate, and so, could safely be discarded. In the interest of detecting a maximum number of valid objects, however, the signal to noise ratio was set close to 3 whenever the data was not so populated by objects as to make the number of false candidates prohibitive. This effect was considered in the bias analysis.

To further increase the efficiency of the SLOSUR program, and reduce the number of potential candidates created, a range circle filter was applied to each potential candidate detected by the program. These range circles are circles whose radius is equal to the maximum distance a given Planet X could have moved over the time elapsed between images. Any moving Planet X, previously calculated to be moving no more than 0.02 degrees per day, would have to be located within the perimeter of the range circle of its location in a previous image and moving retrograde across the sky. This filter was necessary given that the 2.7 square degree area of each image represented nearly 10 times the area covered by the full moon in the sky. As a result, there was no possible way for a Planet X candidate to move from one side of the image in night one to the

opposite side of the image in night two. Before revision, SLOSUR considered pairing any moving object from each night to create a potential candidate with a starting and ending position, thus pairing a moving object detected on one side of the image in night one with a moving object on the opposite side of the image in night two, and every other moving object detected in night two's image. With the range circle filter applied, the SLOSUR program could only pair moving objects detected in each night's image if they fell within a range circle centered on the location of the object in the first image.

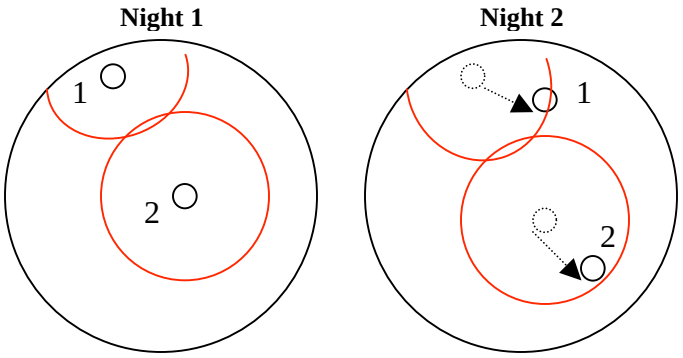

Figure 10: Two objects are shown moving from one location in night one to another location in night two, designated by the arrows shown in night two. The two locations in each night are labeled 1 and 2. Range circles are shown in red.

Figure 10 shows the movement of each object to its new location in the image over two nights. Before the application of range circles, SLOSUR created four potential candidates, two candidates representing each object's actual motion (from location 1 in night one to location 1 in night two, and from location 2 in night one to location 2 in night two), and two false candidates which had moved from location 1 in night one to location 2 in night two, and vice versa. With the range circle filter applied, however, SLOSUR created the two potential candidates which represented each object's true movement, because those were the only pairs which fell within the range circle placed around each moving object in night one.

With the application of these filters into the SLOSUR program code, the efficiency of the program was greatly increased. Whereas the program produced thousands of potential candidates with each review before it was revised, the revised version of SLOSUR yielded less than 200 candidates on average. In addition, the time required for SLOSUR to process one region was effectively reduced, from 24 hours per region before revision, to a matter of 2 to 3 minutes after revision. This was caused in large part by the removal of the onerous requirement for SLOSUR to compute preliminary orbits for all the potential candidates it had detected. These orbits did not need to by computed for every potential candidate produced, as many of them remained false candidates. Instead, these orbits were calculated individually upon the follow-up examination conducted for only those objects that pass visual inspection.

Chapter 3: Results: Preliminary Sky Coverage and Objects Detected

The data collected for this survey was acquired between March 2003 and March 2006. During this period, 3930 regions were surveyed, each of a separate region of the sky. This yielded a total of 10611.0 square degrees of raw sky coverage. This coverage was more tightly concentrated on the ecliptic than originally foreseen, as the Spacewatch group changed their method of data collection over the 2005-2006 observation season to include more saturated revisit coverage nearer the ecliptic, resulting in sparse coverage at higher inclinations. Given a region of 14400 square degrees in which Planet X could reside (assuming a band width of 10 degrees), our raw sky coverage gives approximately 73.6% coverage of this region, neglecting time effects. The preliminary coverage is shown in Figure 11 below.

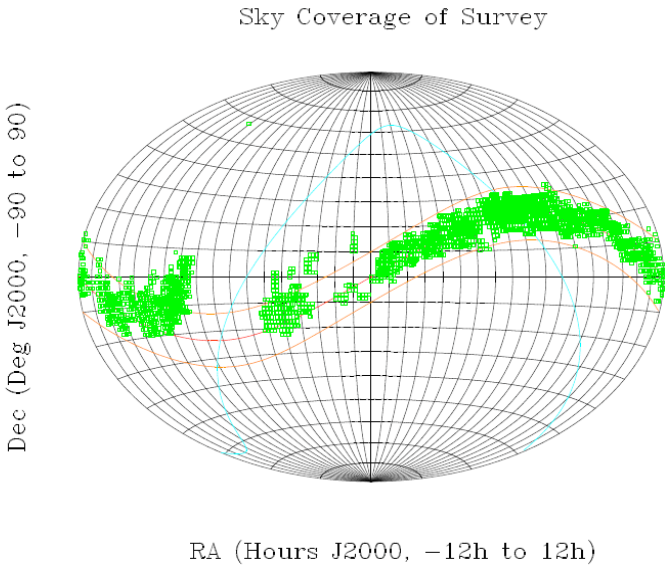

Figure 11: A plot of the completed sky coverage as from March 2003 to April of 2006. The red line represents the ecliptic, the blue line represents the galactic plane, and the yellow lines define the region within 10 degrees of the ecliptic. Each image represents 2.7 square degrees of sky and is shown by a true to scale green box.

The non-surveyed areas shown in Figure 11 represent regions of sparse coverage. The three main holes were caused by the location of the axis of the Milky Way galaxy, the local monsoon season in Tucson, Arizona, where the data was taken, and the change in survey pattern to match the ecliptic very tightly. The absence of any images near the intersection of the blue line (which represents the galactic plane) and the red line (which represents the ecliptic, or plane of the solar system) was caused by the fact that this intersection is in the direction of the galactic center in the plane of the galaxy. The galactic center is a highly populated, dense region of stars and dust in the plane of the galaxy. This survey was not affected by the absorption by that dust, as it was concerned only with relatively close objects located within the solar system, whose light does not travel far enough to be sufficiently absorbed before reaching Earth. However, the number of stars located in these areas of the sky tends to make for extremely dense fields, as a majority of the visible objects in the Milky Way are located along the galactic plane. The vast number of stars in these regions, therefore, hampered the detection of any moving objects. As a result, it was not feasible to see objects when looking toward the galactic center which has the greatest concentration of objects. While difficult, it was possible to view objects in the opposite direction, which points away from the galactic center and is sparser as a result. The other main region of sparse coverage was caused by the local climate at Kitt Peak National Observatory, which experiences an annual monsoon season from June to late July or early August. During this time, the inclement weather bars astronomical observation, resulting in predictably sparse coverage during these times for all visual survey groups at the observatory. This year's monsoon season was of greater duration than normal, however, further extending the period where no data could be gathered. Finally, the Spacewatch group changed their methods of data collection, abandoning their traditional approach to favor coverage that was more tightly focused on the

ecliptic. While allowing them to more efficiently detect and follow up Near Earth asteroids, this resulted in sparse imagery of regions at higher inclination.

Taking into account the atmospheric conditions during the period of data collection to determine the seeing conditions, the Full-Width Half-Maximum (FWHM) was computed for each review. Seeing is an astronomical term for the distortion of images due to atmospheric conditions, such as turbulent mixing of air masses. Instrumentation conditions caused by telescope focus and the quality of optics used can introduce further distortions. Seeing is calculated as the mean of the FWHM, which is a measurement of the width or diameter of a stellar profile (called the 'seeing disk') at half the peak pixel's light intensity. Astronomical objects are not point sources, a consequence of telescopic diffraction and the atmosphere, and so the profile of their image follows a distribution function which is largely Gaussian in nature. To determine the width of this distribution, which theoretically extends infinitely albeit at very low values, the width of the profile is measured at a value equal to half the maximum of the distribution. This width, generally measured in arcseconds ($\frac{1}{3600}$ of a degree each), is the FWHM. Figure 12 shows the FWHM values for the reviews conducted in this survey, and the number of reviews that contained that particular value.

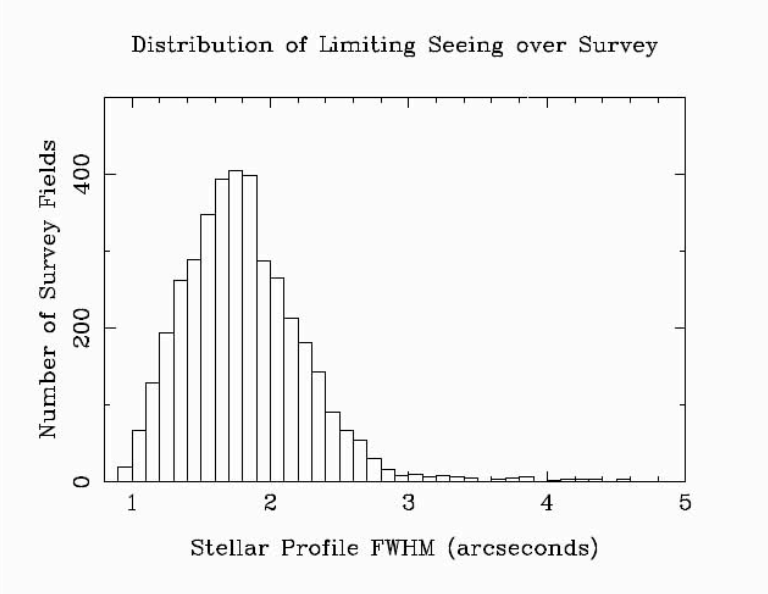

Figure 12: This plot shows the stellar FWHM vs. the number of reviews containing that value of FWHM. Though there were a majority of fields with a low FWHM, corresponding to good seeing conditions, some reviews of poorer seeing were included. This was done to allow for the detection of any bright candidates that might occupy those regions.

Given that 3930 regions were processed over the survey, and with each review requiring six images taken over two nights, the survey involved a total of 23,580 images of 81.3 MB size, giving a total of 1.92 TB of raw data collected directly processed and 1 TB of auxiliary imagery collected. Over the course of the survey, SLOSUR detected a total of 1.37 billion objects, which were reduced to only matched objects (shown by Figure 13), and then reduced further through the applied motion filters, to produce 434,996 candidates needing visual validation.

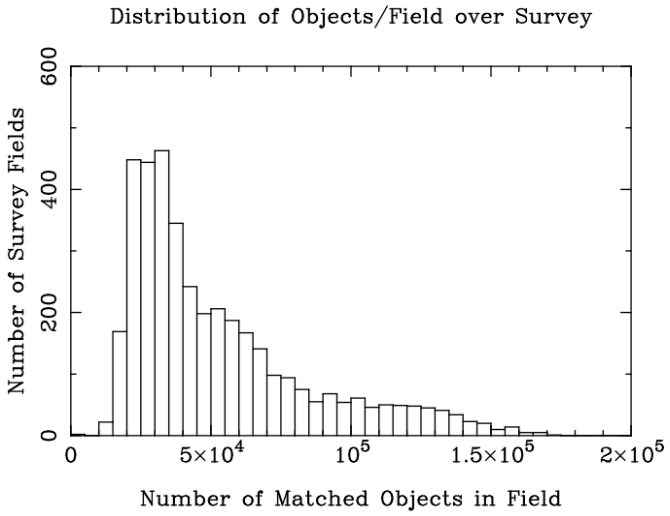

Figure 13: This is a plot of the number of matched objects in a field vs. the number of fields containing that number of objects. SLOSUR needed to search through an average of 30,000 matched objects per region to find any moving candidates, with some regions (primarily taken in the galactic plane) containing up to 150,000 matched objects.

Of these candidates, 668 required in-depth evaluation, after which 23 turned out to be valid detections. The breakdown of detections is shown in Table 1.

Number of Candidates	Cause for Detection	Description of Cause
79	Weather/focus	Star disappears one night because of cloudy weather (poor seeing) or bad telescope focus
304	False positive	Objects such as novae, variable stars, Aphelic Near Earth Asteroids, and real stars that failed the matching criteria
49	Planted objects	Fake candidates planted to calculate the efficiency of the observer
17	Known objects	These were real Kuiper Belt objects that had already been discovered
6	2003 MW12	Our discovery was repeatedly detected in data throughout the survey
106	Stationary point asteroids	Where an asteroid at the stationary point was paired with a non-moving object to create a slow moving candidate
1	Double stationary pairing	Where two different stationary point asteroids were paired together to make a slow moving object
30	Short baseline	Follow-up reviews with baselines of < 30 min., where fast moving objects did not have enough time to move sufficiently to by filtered out
11	Astrometry matching error	An erroneous calculation that caused a stationary object to receive differing location coordinates on one night
30	CCD artifacts	Flaws in the CCD caused by hot pixels, dead pixels, cosmic rays, and cross-talk (induced charge)
17	Single position objects	Contain a single position, which could be a real candidate, but has not been confirmed
2	Non-existent	Recorded as detections erroneously by the observer
16	Unknown	Archived reviews whose data was either burned onto a corrupted DVD or lost

Table 1: The table shows a breakdown of the 668 detection is made throughout the survey. The cause for each detection, and an explanation for that cause is provided.

Out of the 23 valid detections made during the survey, one of those objects was a new discovery. Named 2003 MW12, it was first discovered in January of 2006, but was not confirmed until it was re-imaged in March 2006 by the Spacewatch 1.8 meter telescope (shown in Figure 14). This observation was made possible by a follow-up observational visit to Kitt Peak National Observatory by Eric Roe and Professor Jeffrey Larsen of the United States Naval Academy. Since its confirmation, 2003 MW12 has been observed by other astronomical survey groups as well, such as the Near Earth Asteroid Tracking survey at Jet Propulsion Laboratory (JPL), who found it on images from May 30, 2002 and June 7, 2002 by stacking survey images.

Figure 14: The 1.8 meter Spacewatch telescope located at Kitt Peak National Observatory, outside of Tucson, Arizona. It is a f/2.7 telescope (meaning that the ration of the focal length of the telescope to it's aperture, or diameter, is 2.7) with the CCD placed at prime focus. Follow-up images of 2003 MW12 were made using this telescope, the first to be taken since its discovery 3 months prior, confirming the objects status as a valid Kuiper Belt object. Picture taken by Professor Jeffrey Larsen of the United States Naval Academy.

2003 MW12 has a semi-major axis of 45.9 AU, which equates to 6.9 billion miles from the Sun, whereas Pluto orbits at 5.9 billion miles from the Sun. The great distance with which 2003 MW12 orbits the Sun results in an orbital period 310.4 years, and results in an apparent motion of only 2.66 arcseconds per hour, equivalent to moving the diameter of a dime when held a full kilometer away every hour. Its great distance and absolute magnitude (H) of 3.8 combine to give 2003 MW12 an apparent magnitude (V) is 20.5. This is approximately as bright as a 100 watt light bulb would be if placed on the surface of the moon and viewed from Earth. The candidate image where 2003 MW12 was first discovered is shown in Figure 15, and follow-up images of 2003 MW12 moving over a 45 minute period are shown in Figure 16.

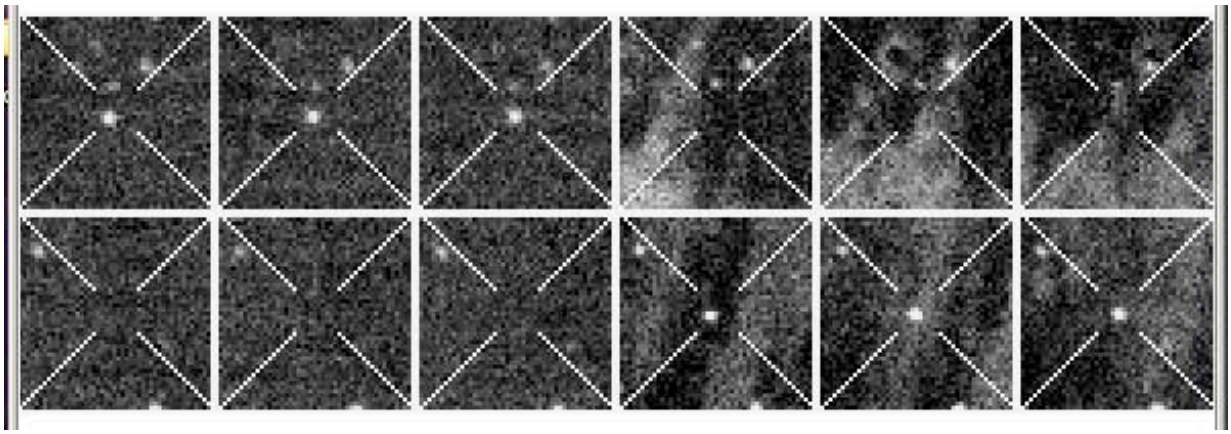

Figure 15: This is the discovery image of 2003 MW12, as seen in the review. 2003 MW12 be seen in the top left 3 boxes, and the bottom right 3 boxes, representing its positions on two nights. The empty boxes show the object had vacated the positoin on the other night.

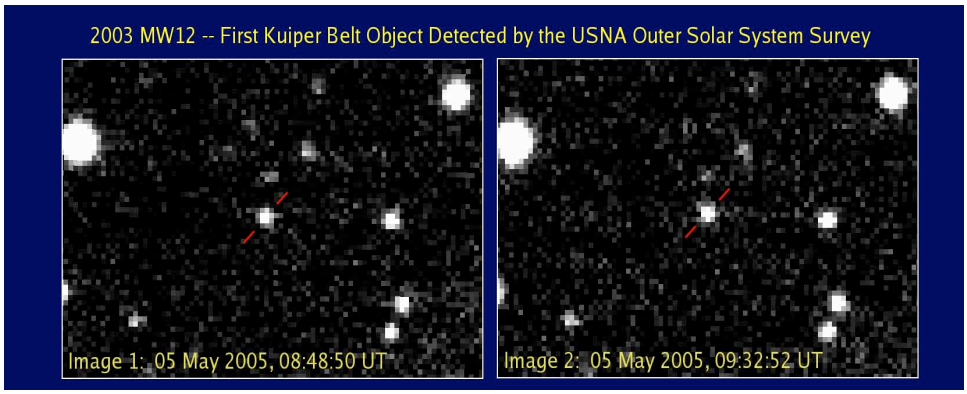

Figure 16: Two images of 2003 MW12 (highlighted by red lines) taken 45 minutes apart. The motion of the object is barely discernible over this time frame, as the object has moved only 2 arcseconds slightly up and to the right.

To determine the size of 2003 MW12, assumptions were made regarding it's albedo. Albedo is the percentage of incident light that an object reflects, with 1 being a mirror and 0 being an object with no reflectivity, such as a black hole. Albedos of rocky objects in the solar system, such as Mars, the Moon, and many asteroids, range from 0.05 to 0.25, while icy bodies have higher albedos, such as Enceladus, an icy moon of Saturn with an albedo of 0.8 and Europa,

an icy moon of Jupiter with an albedo of 0.6. Assuming that 2003 MW12 is predominantly rocky in composition, an albedo of 0.089 was estimated, and using Equation 6,

$$D(km) = \left(\frac{1329}{\sqrt{\Lambda}}\right)\left(10^{-0.2H}\right) \qquad \textbf{Eq. 6}$$

where Λ is the albedo and H is the absolute magnitude, the diameter (D) of 2003 MW12 was calculated to be approximately 770 kilometers.[17] This size means that 2003 MW12 is a little less than one third the size of Pluto, and currently the 10th largest known Classical Kuiper Belt object. Table 2 describes the 11 largest known Kuiper Belt objects.

Absolute Magnitude	Diameter (km)	Name	Semi-major axis (a)	Eccentricity (e)	Inclination (i)
-1.2	3000	2003 UB_{313}*	67.67	0.441	44.181
-0.4	1500	2005 FY9	45.707	0.155	28.998
1.6	1700	(90377) Sedna	582	0.844	11.935
2.3	1530	(90482) Orcus	39.386	0.219	20.581
2.6	1260	(50000) Quaoar	43.547	0.035	7.989
3.3	790	(55565) 2002 AW197	47.37	0.131	24.379
3.6	840	(55637) 2002 UX25	42.525	0.142	19.49
3.7	970	(20000) Varuna	42.954	0.051	17.192
3.8	770	2003 MW12	45.904	0.137	21.493
3.8	770	2002 MS4	41.864	0.14	17.67
3.9	1200	(84522) 2002 TC302	55.037	0.291	35.119
4.5	900	(19308) 1996 TO66	43.16	0.121	27.479

Table 2: The largest Kuiper Belt objects discovered. 2003 UB_{313} is marked with an asterisk because it is a highly inclined scattered Kuiper Belt object.

Other objects detected by the survey are shown in Table 3 and represent known Kuiper Belt objects.

[17] Bruton, Dan, "Conversion of Absolute Magnitude to Diameter for Minor Planets," <http://www.physics.sfasu.edu/astro/asteroids/sizemagnitude.html>, 1 February 2006.

Object Name	Measured Absolute Magnitude (H)	Number of times detected
(20000) Varuna	3.7	1
2003 MW12	3.8	6
(19521) Chaos	4.9	3
(48639) 1995 TL8	5.4	2
(26308) 1998 SM165	5.8	3
(88269) 2001 KF77	6	1
(4971) 2000 KK4	6	1
(33340) 1998 VG44	6.5	1

Table 3: Names, absolute magnitudes and the number of times each known Kuiper Belt object was detected in the survey.

Although they are already known, the detection of these Kuiper Belt objects still yields information about the survey. The detection of an object such as (26308) 1998 SM165 is significant because, with an absolute magnitude of 5.8, it has an apparent visual magnitude of 21.4. Any Planet X candidate would be brighter than 21.4 magnitudes, and therefore, should have been detected by the survey had it been located in any of the regions observed. The detection of even more faint objects, such as (88269) 2001 KF77, (4971) 2000 KK4, and (33340) 1998 VG44 continues to emphasize this point. Given these detections, this survey should have been able to detect an object with the magnitude of 2003 MW12 out to 100 AU.

Chapter 4: Measuring Survey Efficiency

Determining the efficiency of the survey is important, not only to extrapolate conclusions from the collected imagery data, but also for the bias needed to determine the true spatial coverage of the project. The overall efficiency of the survey included the efficiency of both the SLOSUR analysis program in detecting moving objects of given orbital properties and the observer's efficiency at identifying valid detected motions produced by SLOSUR. To measure both of these efficiencies, test motions were generated by stripping real objects out of images and planting them in such a way that they would represent valid moving objects. The properties of these planted test motions could be altered at will, after stripping them from the original image, such as dividing the magnitude of the object by 2.5 to reduce the object's brightness for each magnitude. An example of planting a test motion into an image is shown in Figures 17 and 18.

Figure 17: A portion of an actual image from the data collected for this survey.

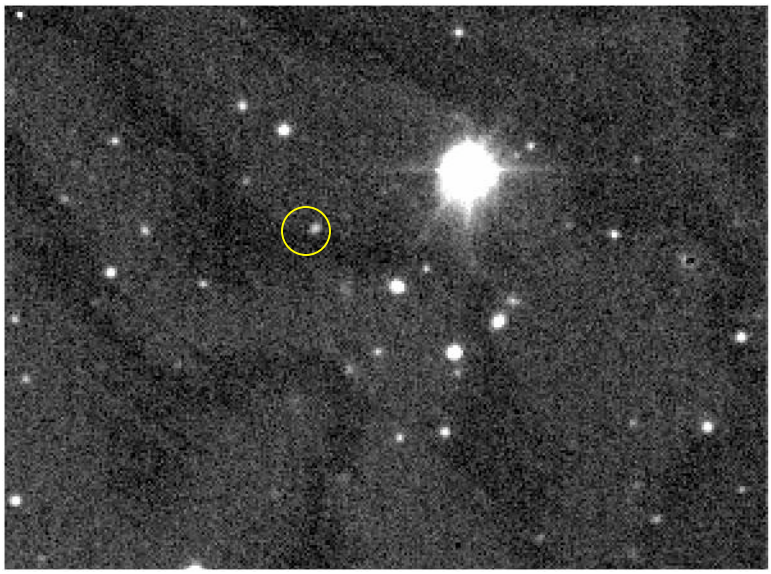

Figure 18: The same image as Figure 17, but with a planted object inserted (denoted by a yellow circle).

To test SLOSUR's full range of efficiency, the absolute magnitude of each object was varied among a wide range of magnitudes, from those known to be below SLOSUR's limits of detection to those known to be easily detectable. In all, 167 of these objects were created, planted into 30 selected images so as to give a range of rates, and analyzed during the survey. The 30 images were chosen such that the clutter (or density of the field) was varied among the images, with the extremes represented. Those images were then analyzed by SLOSUR and given to the observer. No knowledge of which reviews would contain the planted objects was given to the observer, allowing the efficiency of the SLOSUR computer software to be impartially determined by comparing the total planted objects with the number of those objects identified by SLOSUR as moving, and the number identified by the observer as valid candidates. These measurements appear in Figure 19, and show that 37% of the planted objects were detected. Figure 19 also shows that while SLOSUR's efficiency declined from approximately 90% at or below objects of 20th magnitude to slightly over 50% for objects of 21st magnitude, to

only 5% for objects of 22nd magnitude (representing the decreased signal of the object), observer efficiency stayed predominantly constant at greater than 90% of the presented candidates. The efficiency was not a function of rate until the object represented orbits with a semi-major axis of 2000 AU or larger. This resulted in a total survey efficiency of greater than 90% for objects at or under 20th magnitude, approximately 50% for objects around 21st magnitude and 10% for objects of 22nd magnitude. Because the images represent typical sky conditions and field crowding, those effects can be considered to be folded into the efficiency calculation.

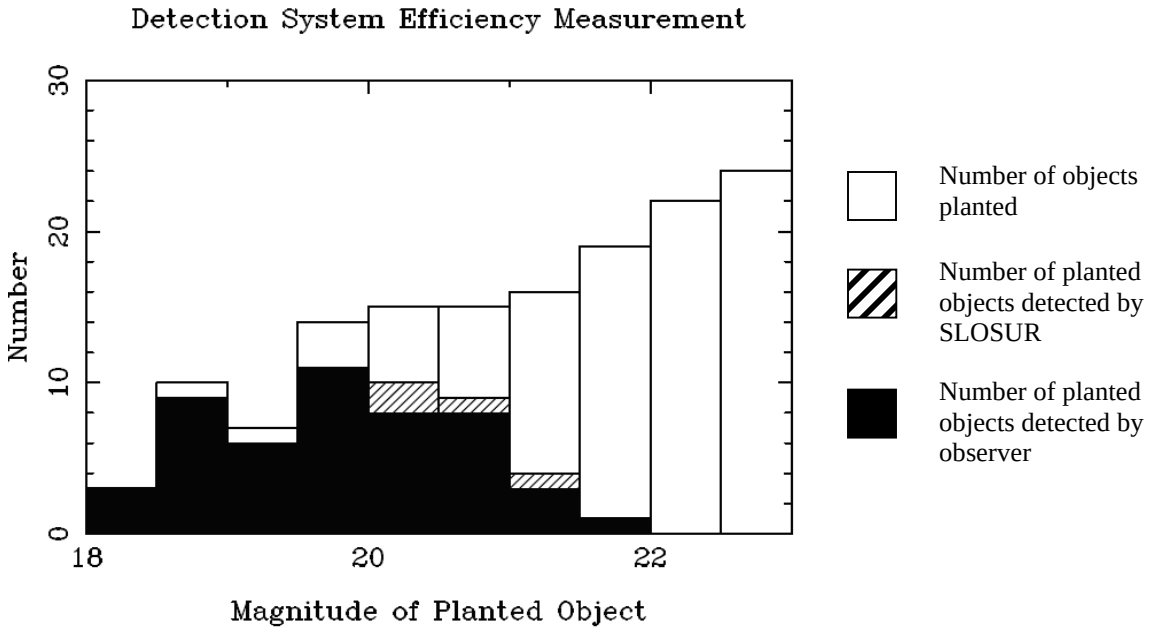

Figure 19: The measured efficiency of the survey, as a plot of the magnitude of object vs. the number of times that object was detected. The efficiency of SLOSUR is found by dividing the hashed column with the open, while the efficiency of the observer is found by dividing the black column with the hashed. The total survey efficiency (as a function of absolute magnitude) is found by multiplying the SLOSUR and observer efficiencies.

Chapter 5: Determining the Bias Model

Another key objective of the project was to determine the unique volume of space searched while looking for Planet X. This calculation is essential to placing firm limits on the existence of Planet X. The determination of the unique amount of space searched during this project, called the bias, was approached via two different methods. The bias is complicated by several effects. First, all possible realizations of Planet X are moving, both as a group, and with respect to each other in position phase space. As a result, only a percentage of the area of each image will be unique to the search, as some hypothetical Planet X orbits from previous images will have moved into the area over the elapsed time period since they were last seen. Furthermore, some images physically overlap due to revisits made by Spacewatch over time. The solution to these problems was to create two computer simulations of a population of objects whose orbital characteristics were consistent with those of Planet X. The computer models were used to show the evolution of this population of Planet X candidates over a specified time period and, from this information and the regions surveyed, to derive the bulk motions of the population. Over the period of this project, the motion of these objects will be used to determine their current location for each image taken, allowing the unique area of each image to be determined.

1. First Approach: Computer Simulation of Volume Searched

The initial approach to calculate the bias for the study involved the creation of several computer programs to simulate the motion of a virtual population of planetary candidates over the time span of the project. This simulation was then used to generate approximations for the total motions of those candidates over time. The population was evolved for each image taken, in the 2.7 square degree region of space covered by that image, and then evolved with the time aging of the field. In this way, possibilities for the motion of individual candidates from one image to the next were covered by the net motion of a population. As a result, any candidates found to be occupying the region of a new image when it was taken represented a region of non-unique space. The amount of non-unique space found in each image was used to calculate the percentage of that image that was unique, thus allowing the calculation of the biased sky coverage for the survey. These percentages were then stored in a 201 X 201 X 201 element dimension array, in which each voxel (a cubic pixel) represented one cubic AU of space in the solar system.

Creating the Bias Model Program

Before the bias computer model could be created, a general idea of the relative motions of these objects had to be developed. First, a simple demonstration model of the object's motion, using Newton's laws and circular orbits, was created. As previously noted, the major force acting on massive bodies in the solar system is described by the law of gravitation, resulting in an object's velocity depending purely upon its distance from the Sun, as shown in Eq. 7. Since

the distance an object moving at constant speed has traveled (*d)* is proportional to the object's velocity (*v*) multiplied by the time elapsed (t), the distance an object has traveled in its orbit is:

$$d = \sqrt{\frac{GM}{r}}t \qquad \textbf{Eq. 7}$$

assuming a circular orbit. However, this distance will be along the path of that object's orbit, which represents a radial arclength around the Sun. With this arclength also being defined by $d = r\theta$, where r is the radial distance from the Sun, and θ is the angle subtended by the arc, an object's location in the plane of the solar system in Cartesian coordinates (*x*,*y*,) can be defined for circular motion as:

$$x = r\cos(\theta) \qquad \textbf{Eq. 8}$$
$$y = r\sin(\theta)$$

These coordinates were used to compute the motion of the population of objects shown in Figure 20. This figure shows identical objects, in orbits from 40 AU from the Sun (the approximate distance of Pluto) to 250 AU from the Sun in 5 AU increments, moving along their respective orbits which have been broken into five 50 year time intervals, each represented by a different color.

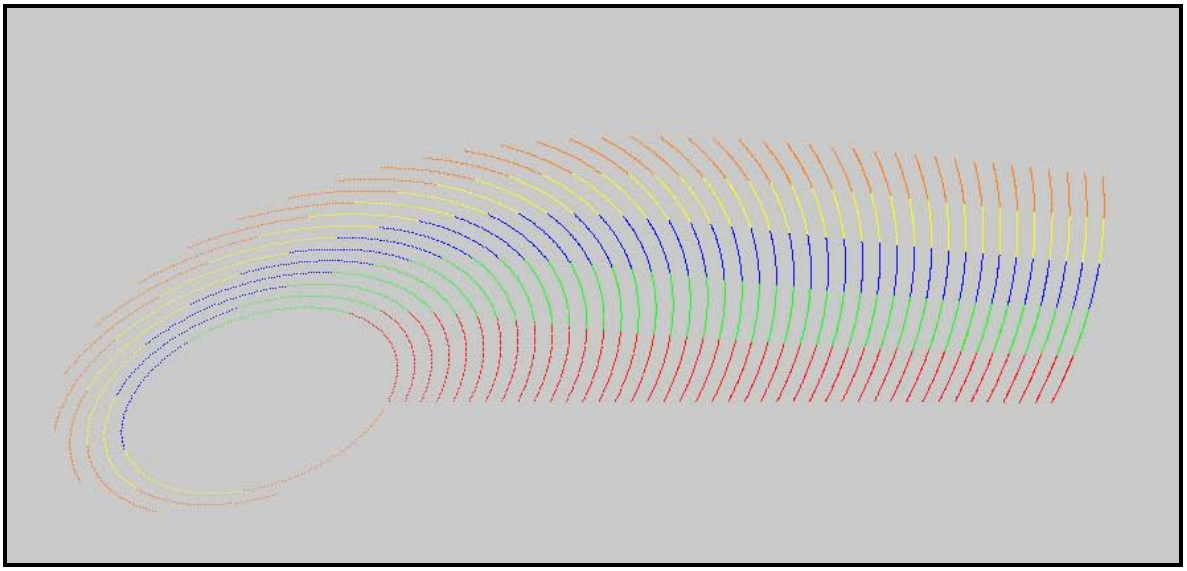

Figure 20: A simplified model of the motion of 42 identical objects along their orbits around the Sun (which is not shown). These orbits start at a distance of 40 AU from the Sun, and move outwards in increments of 5 AU for each object as seen from above.

Figure 20 shows that after the first time interval of 50 years, the objects that are closer to the Sun have moved farther along their orbits than those objects that are farther out, resulting in a spreading of the spatial distribution of those objects which had started at t=0 along a single line of sight. Based on this figure, the bias model should show that the population of closer Planet X candidates becomes more diffuse over time, and that the velocity of those objects increases with decreasing radial distance from the Sun.

The final computer simulation program was written in Perl, chosen for its ease in integration of external programs that would be needed to compute the orbits of each object in the population, and is included in Appendix B. The computer program generated a random population of 10000 objects that have orbital properties within the range of those predicted for Planet X. This population was then sent to an orbital ephemeris program (written by members of the Spacewatch Project and Dave Tholen from the University of Hawaii) to compute their orbits. The ephemeris program returned two data files for each object in the population which specify

that object's location at two different times. The program then extracted the Heliocentric Cartesian coordinates (*x*,*y*,*z*) of the object from each data file, in AU and centered on the Sun, and placed them into lists that become datasets for visualization of the object's movement in space. To ensure that the bias program was working correctly, one of these datasets was set to plot a small portion of each object's orbit using relatively small time increments so that we could examine them. Initially, the ephemeris program returned the ecliptic latitude (λ) and the ecliptic longitude (β) of the object at a given time interval, and these values were used to calculate the location of the object in Cartesian coordinates using Eq. 9.

$$x = r\cos(\lambda)\cos(\beta) \qquad \textbf{Eq. 9}$$
$$y = r\sin(\lambda)\cos(\beta)$$
$$z = r\sin(\beta)$$

What we initially found that was unexpected was that, instead of smooth arcs for each object, the objects appeared to move in loops whose net motion was in the form of an arc. This visualization is shown in Figure 21. Seen close up, as shown in Figure 22, there appear to be an equal number of arcs for each object, and the number of arcs equaled the number of years elapsed over which their positions were calculated. Given this information, it was determined that the loops were caused by the Earth's movement, and not the objects themselves. As seen from the Earth, an object moving slower than the Earth will appear, during some time period within a year, to move in the retrograde direction before resuming its course once again because, during this time period, the Earth passes the object in its orbital revolution. These loops were created by the fact that the ephemeris program had returned geocentric (Earth-centered) ecliptic coordinates (λ, β), rather than heliocentric (Sun-centered) ecliptic coordinates.

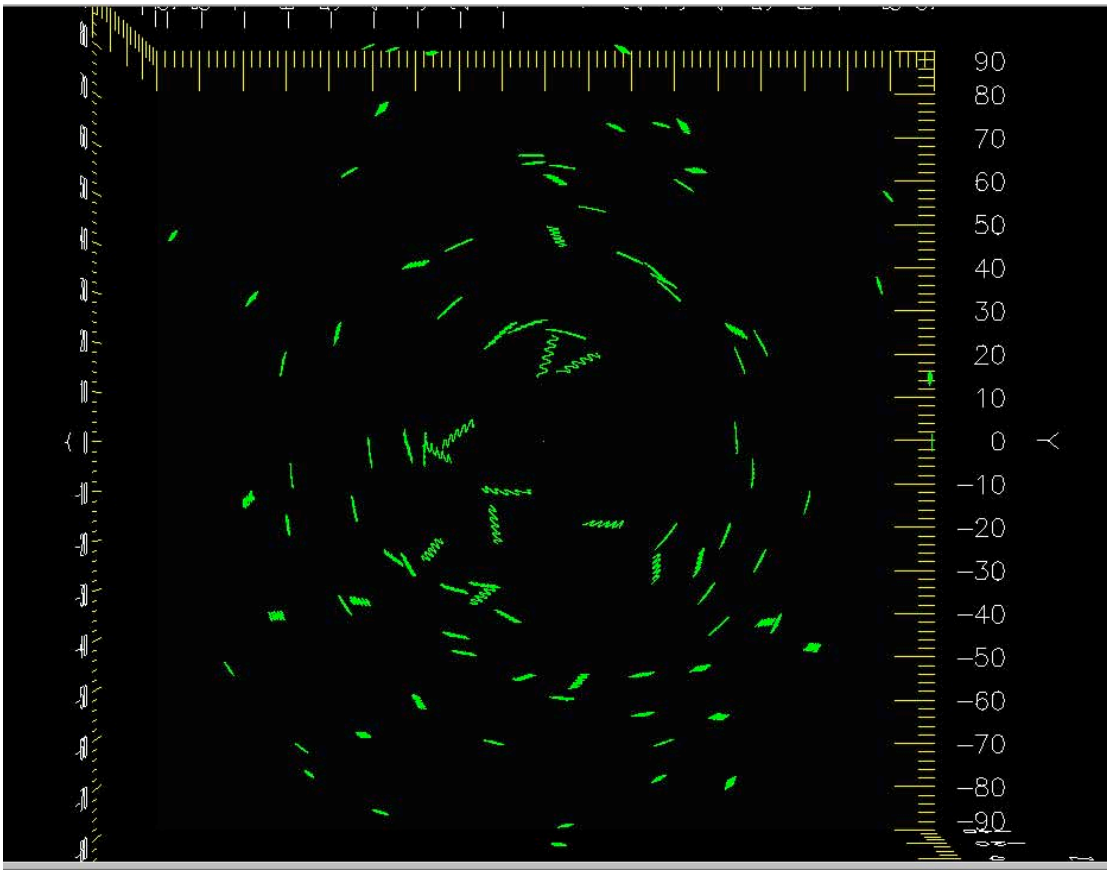

Figure 21: This visualization shows a random selection of 100 objects generated by the bias model program. They are moving about a portion of their orbits over a 5 year time interval, which are centered about the Sun (which is not shown) and seen from above. The axes of this plot are labeled in AU. The smooth arcs in which each object should move appear to be loops whose net motion follows an arced path.

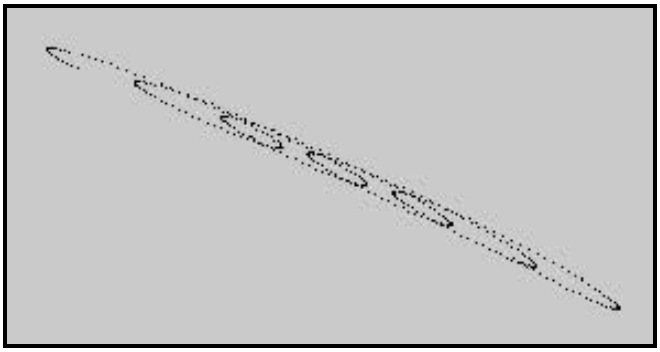

Figure 22: A close up look of one object moving along its orbit over a time interval of 5 years. Instead of a smooth arc, there are 5 loops present.

To correct for this condition, the program code was altered to directly produce heliocentric Cartesian coordinates, therefore bypassing the need for the coordinate transformation from ecliptic to Cartesian coordinates. After being re-run, the orbit path dataset showed the smooth arcs depicted in Figure 23. Given that each object moves in a smooth arc along its orbit, the ephemeris program was working correctly.

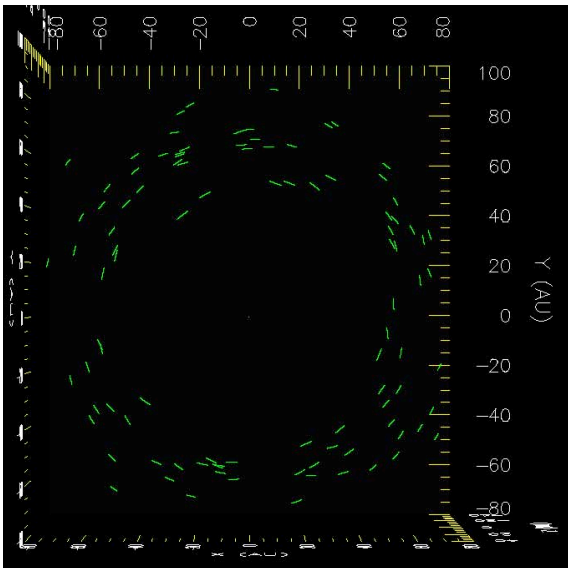

Figure 23: A new dataset of 100 objects moving along their orbits that shows smooth arcs.

With the accuracy of the population model demonstrated, the program was re-run to generate the datasets used for the bias model visualization. In this visualization, locations of the 5000 objects generated were recorded at the beginning and ending of 5 one year time steps. To place all of the objects along an initial line of sight (a complicated and artificial thing to arrange using orbital elements), the relative motion from their initial location was plotted, rather than their actual positions in space. To do this, each object was placed along one axis, at a radial distance equal to the radial distance of their initial location in space. Next, their second location after one year was computed by the ephemeris program. The difference between this location and the initial location in phase space was taken to get the relative motion vector between the two points, and

this relative motion vector was plotted from the object's starting point along the line of sight. Therefore, the resulting datasets do not show the actual location of each object in phase space, but rather the relative change in their position from some common line of sight at time t=0 to a new location after each year has passed. The final population motion visualization is shown in Figures 24 and 25.

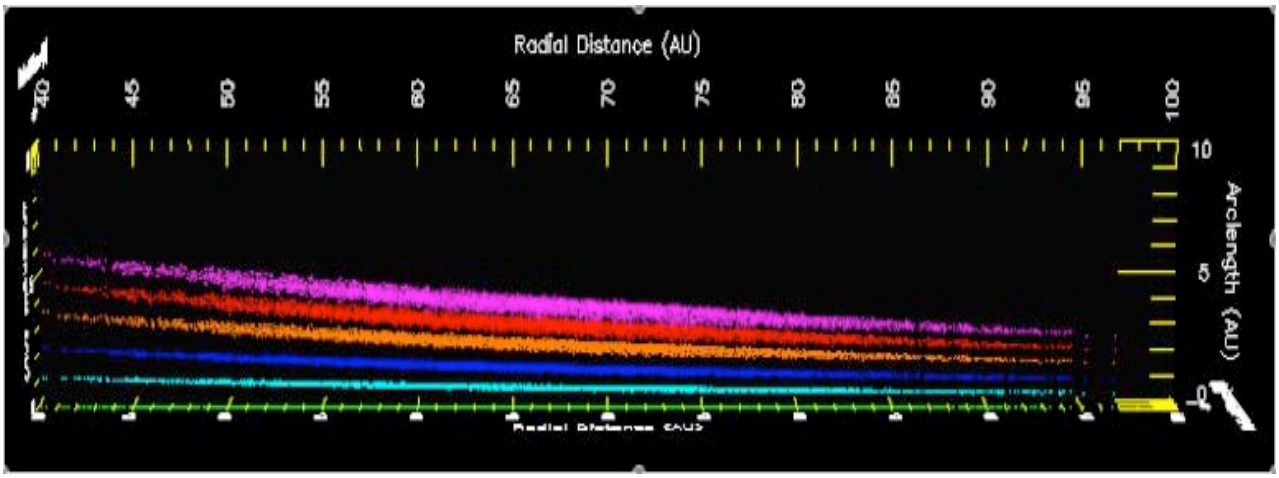

Figure 24: A visualization of 5000 Planet X candidates moving along their orbits over a 5 year time period, as seen from above. The radial distance is from the Sun, with the closer objects on the left side of the image. The 5 year period is divided into 1 year increments, each shown as a different color.

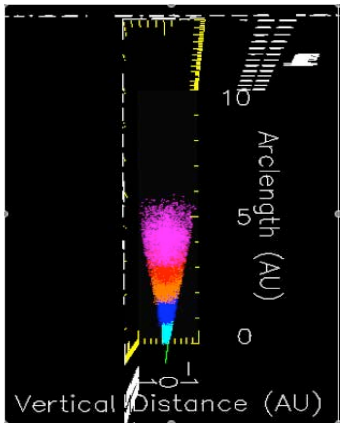

Figure 25: This is the same distribution as Figure 17, but is seen from the end, looking towards the Sun. Here the object's movement out of the plane of the solar system, labeled the vertical distance, can be seen.

With this model, the mean motion of the population is shown, as is the motion of each individual object with respect to that mean motion. This model confirms our earlier claim that the population of objects would become more diffuse over time, with the objects closest to the Sun moving farther and spreading out the most. It also confirms that hypothetical populations of Planet X candidates do move with well defined behavior, and that each candidate in the population shows individual deviation from that behavior. This simulation shows our two main biases are against faint and closer, fast moving, Planet X candidates, making them harder to detect.

Later, the computer modeling program was rewritten to generate a 10,000 candidate population. The larger number of candidates would further increase the accuracy of the simulated population. Taking into account the timespan of the data, the program was also revised to run from May, 2003 to May, 2006. During this time period, the location of all 10,000 objects was calculated on the first day of each month, and written to an output file for that month, resulting in 36 files containing the location of each object in Cartesian coordinates.

Analysis of Modeled Data

After accurately simulating 10,000 planetary candidates, the model needed to be analyzed to determine their motion, both as individual objects and as a population. A data analysis program was written to accomplish this task, essentially calculating the motions shown in Figures 24 and 25. Also written in Perl, this program took the output files from the computer simulation, which consisted of one file for each month, and contained the starting and ending location for each object of that month. These files were first organized into object files, which contained only one candidate, and recorded its location for every month in the modeled time

frame. The point of this was to separate each object from the others by making it a separate file. This made it easier to work with, as objects could be tracked from month to month simply by moving from line to line in the file. More importantly, this allowed for the objects to be sorted into bins depending on their distance from the Sun.

The creation of bins, in which to group the objects, was necessary to determine the average motion of the population. Because the motion of each object is dependent on its radial distance from the Sun, averaging the motion of every object in the population to determine an average rate would be inaccurate, as the population ranged in radial distance from approximately 45 to 95 AU from the Sun. The calculation would need to be broken down into smaller increments, along a radial distance axis, to allow for the different motions of an object at different radial distances. Ideally, these increments would be infinitesimal in size. However, this would require intensive computing power and was not necessary for this project, as the array built to hold the bias coverage calculations for each image contained voxels of one AU^3. Increments of any size smaller than one AU would surpass the resolution of this array, and would be combined with neighboring increments until the one AU pixel increment was reached. Given this minimum resolution, the object bins were divided into one AU increments along the radial axis, starting at 35 AU and ending at 99 AU, in order to ensure that all objects were included.

Initially, the objects were sorted into bins for each month, as it was easier to program, but this allowed objects to 'migrate' from bin to bin when their radial motion was greater than 1 AU towards or away from the Earth. The constant migration of objects between bins flawed the ensuing calculations, resulting in values whose standard deviations were greater than the average motion of the population in that bin. This was solved by sorting the objects into bins based

solely upon their initial radial distance, a method which allowed no migration. The final distribution of objects into bins is shown in Figure 26.

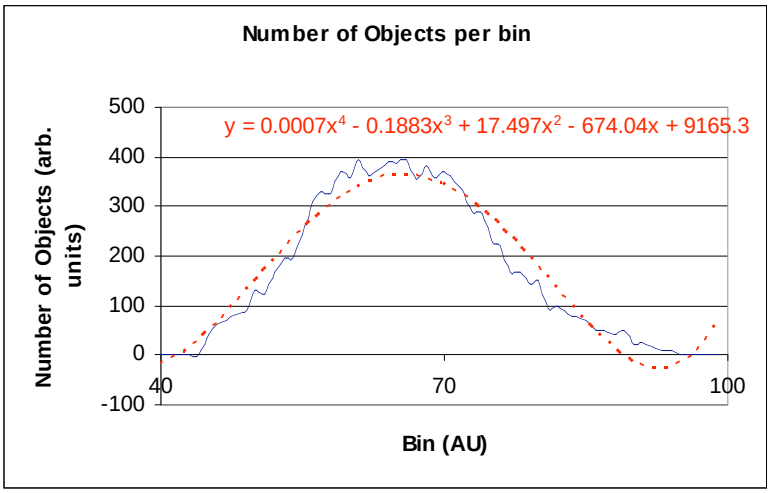

Figure 26: Bin distance v. the Number of Objects in that bin. Objects were sorted into bins based on their initial radial distance from the Sun. Shown superimposed on this distribution is a fourth order polynomial trend line, which gives the approximate number of objects (y) for any given bin (x).

To keep track of which object was in each bin, a file was made for each bin, and the object number (out of 10,000) was written into each bin file. With each bin treated separately, the objects sorted into that bin were used to calculate the mean motion and standard deviation for the bin, yielding values for the average motion of a candidate at a given radial distance (with one AU resolution). Calculating these motions required opening the bin file, reading the objects in that bin, opening their separate object files one at a time, and storing their monthly locations into different arrays. An array was created for each variable needed to compute the mean and standard deviation for the bin. After reading all of the objects into each array, the mean was calculated using Eq. 10, and the standard deviation from Eq. 11.

$$mean(x) = \left(\frac{\sum x_i}{N_x} \right) \qquad \textbf{Eq. 10}$$

$$stddev_x = \sqrt{|\mathrm{var}_x|} \qquad \textbf{Eq. 11}$$

$$\text{where } \mathrm{var}_x = \sum x_i^2 - \left[\frac{\left(\sum x_i\right)^2}{N_x}\right]\left[\frac{1}{N_x - 1}\right] \qquad \textbf{Eq. 12}$$

and N_x is the number of values summed for that location in the array.

Creating the Line of Sight Program

A line of sight program was then created to calculate the unique sky coverage for the survey. The program accomplished this by creating a 201 X 201 X 201 element array to model the region of sky in which Planet X could reside, with each voxel (cubic pixel) in the array representing 1 AU^3, and the Sun placed at element 100 X 100 X 100. Each image, then, could be modeled as a volume cone extending from space in the direction of the image's coordinates in the sky. Because each image covered an area of 2.7 square degrees in the sky, the opening angle of this volume cone would be 0.032 steradians in solid angle. The coordinates for the volume cone are affected by the Earth's position in it's orbit at the time the image was taken, and by the coordinates of the image itself. The position of the Earth, in heliocentric Cartesian coordinates, can be found with sufficient precision from Equations 13, 14, and 15.

$$x_e(t) = r\cos\left[\omega\left(t - t_0\right)\right] + x_o \qquad \textbf{Eq. 13}$$

$$y_e(t) = r\cos\left[\omega\left(t - t_0\right)\right] + y_o \qquad \textbf{Eq. 14}$$

$$z_e(t) = z_0 \qquad \textbf{Eq. 15}$$

where (x_0, y_0, z_0) is the position of the Sun at a given time and $r = 1$ because the Earth's semi-major axis is 1 AU. Taking into account the direction in which the image was looking,

determined from its coordinates in Declination (Dec) and Right Ascension (RA), the position of the line of sight in heliocentric Cartesian coordinates is given by Equations 16, 17, and 18.

$$x_{los}(r,t) = a_x r + x_e(t) \quad \textbf{Eq. 16}$$

$$y_{los}(r,t) = a_y r + y_e(t) \quad \textbf{Eq. 17}$$

$$z_{los}(r,t) = a_z r + z_e(t) \quad \textbf{Eq. 18}$$

Here the slopes (a_x, a_y, a_z) are related to the radius of the line of sight, the galactic latitude (λ) and the galactic longitude (β). Using these variables, the slopes are defined as:

$$a_x = r\sin(\beta) \quad \textbf{Eq. 19}$$

$$a_y = r\cos(\beta)\cos(\lambda) \quad \textbf{Eq. 20}$$

$$a_z = r\cos(\beta)\sin(\lambda) \quad \textbf{Eq. 21}$$

Being able to calculate the volume cone for the line of sight, a line of sight was placed for each review in the survey. The pixels within that volume cone were identified by a sphere at a distance r along the line of sight, whose radius was R. The radius of this sphere (R) could then be described as:

$$R = \alpha r \quad \textbf{Eq. 22}$$

where $\alpha = 0.016$, one half of the opening angle of the volume cone. All of the voxels within $R + 0.5$ of the center of that sphere, located on the axis of the line of sight, had some percentage of their volume located within the volume cone. Given the value of R for any particular r, then, the ranges of voxels located within the volume cone are given by Equations 23, 24, and 25.

$$x_{los} - (R + 0.5) - 1 \le x \le x_{los} + (R + 0.5) + 1 \quad \textbf{Eq. 23}$$

$$y_{los} - (R + 0.5) - 1 \le y \le y_{los} + (R + 0.5) + 1 \quad \textbf{Eq. 24}$$

$$z_{los} - (R + 0.5) - 1 \le z \le z_{los} + (R + 0.5) + 1 \quad \textbf{Eq. 25}$$

The number of voxels in the region of any volume cone where Planet X could reside can then be found by moving along the line of sight from $r = 45$ AU to $r = 75$ AU. This was done in the line of sight program using 0.5 AU increments, as the resolution of the array was 1 AU per side of the voxel.

With the number of voxels for each line of sight determined, the net motion and dispersion of the generated population of planetary candidates was used to calculate the fraction of each voxel that could have previously been observed. This was done by placing the generated population along the line of sight for every region in the survey, just as the generated population was placed along a common line of sight in the computer simulation shown in Figure 24, and evolving that population over the length of the survey. For each region, therefore, the locations of the generated objects from preceding regions could be determined and compared to the voxels known to be within the current volume cone. If any objects were found to be occupying this volume, the location they occupied was considered non-unique space, a region containing objects that had previously been observed. This location was then removed from the running total of unique sky coverage for the survey. This method required keeping track of each population, from its original placement through the end of the survey, done through arrays, and moving the population with the values calculated for the net motion and dispersion as a function of distance. Therefore, with each new review, the populations for preceding reviews needed to be shifted to their new locations, and dispersed. A computer model was built to visualize the results of this process, and is shown in Figures 27, 28, 29, and 30. In this model, the solar system is placed for scale, although the orbits for the planets are approximated to be circles, and new lines of sight are placed every month for the reviews covered during that month. At the outset, shown by Figure 27, there we no reviews completed. However, Figure 28 shows that, as time progressed

and reviews were completed, lines of sight were placed in the model. With the addition of more reviews, shown in Figure 29, the previous lines of sight became 'smeared,' as their corresponding populations became more dispersed. Finally, by Figure 30, at the end of the survey, lines of sight have started to overlap, creating areas of greater and lesser coverage.

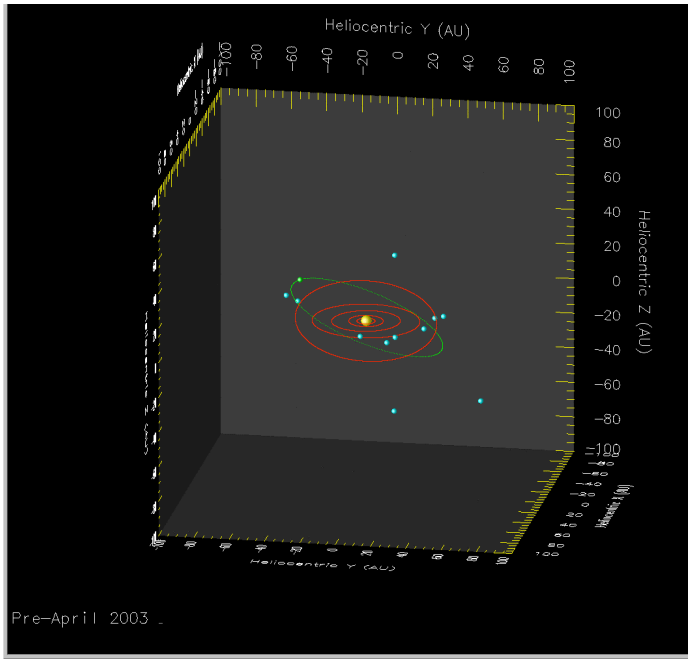

Figure 27: The bias model, as a 201 X 201 X 201 element array, prior to the addition of any data, which began in April, 2003. The solar system is depicted to give a sense of scale, although the orbits for each planet are modeled as circles, rather than ellipses, for simplicity. The Sun is shown as the yellow ball at position 100 X 100 X 100 (the middle of the array), and is disproportionately large for easy recognition. The orbits of the planets that can be seen are Jupiter, Saturn, Uranus, Neptune, and Pluto, and are shown in red. 2003 MW12, discovered by this survey, is shown in green, and other known large Kuiper Belt Objects, detected by this survey, are shown as blue spheres.

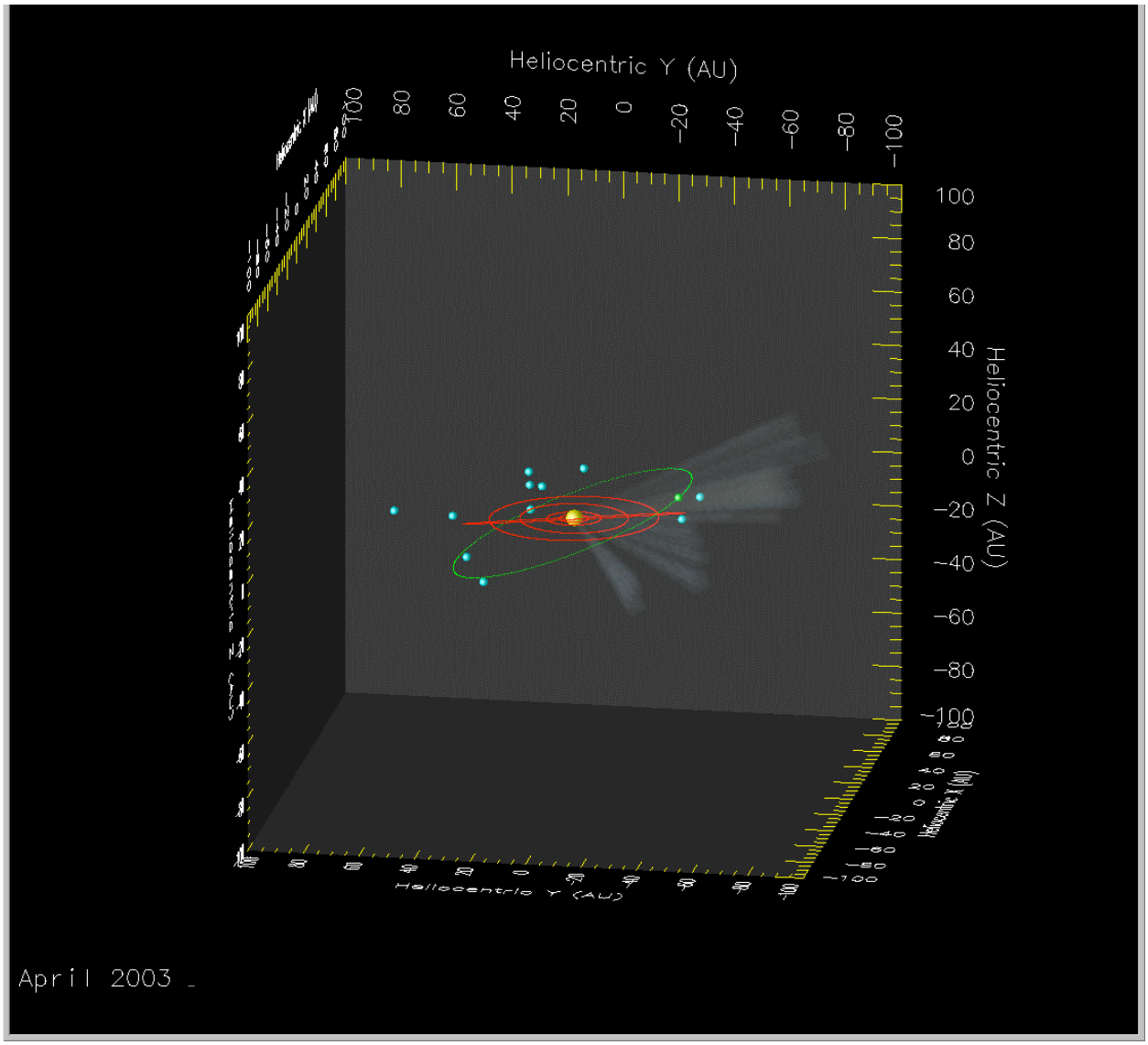

Figure 28: The first month of the survey, April, 2003, where the lines of sight from reviews gathered during that month have been added to the array. Note that because these lines of sight extend up to 100 AU from Earth, the position of Earth in its orbit has a maximum effect on their direction or location to distances of only 2%.

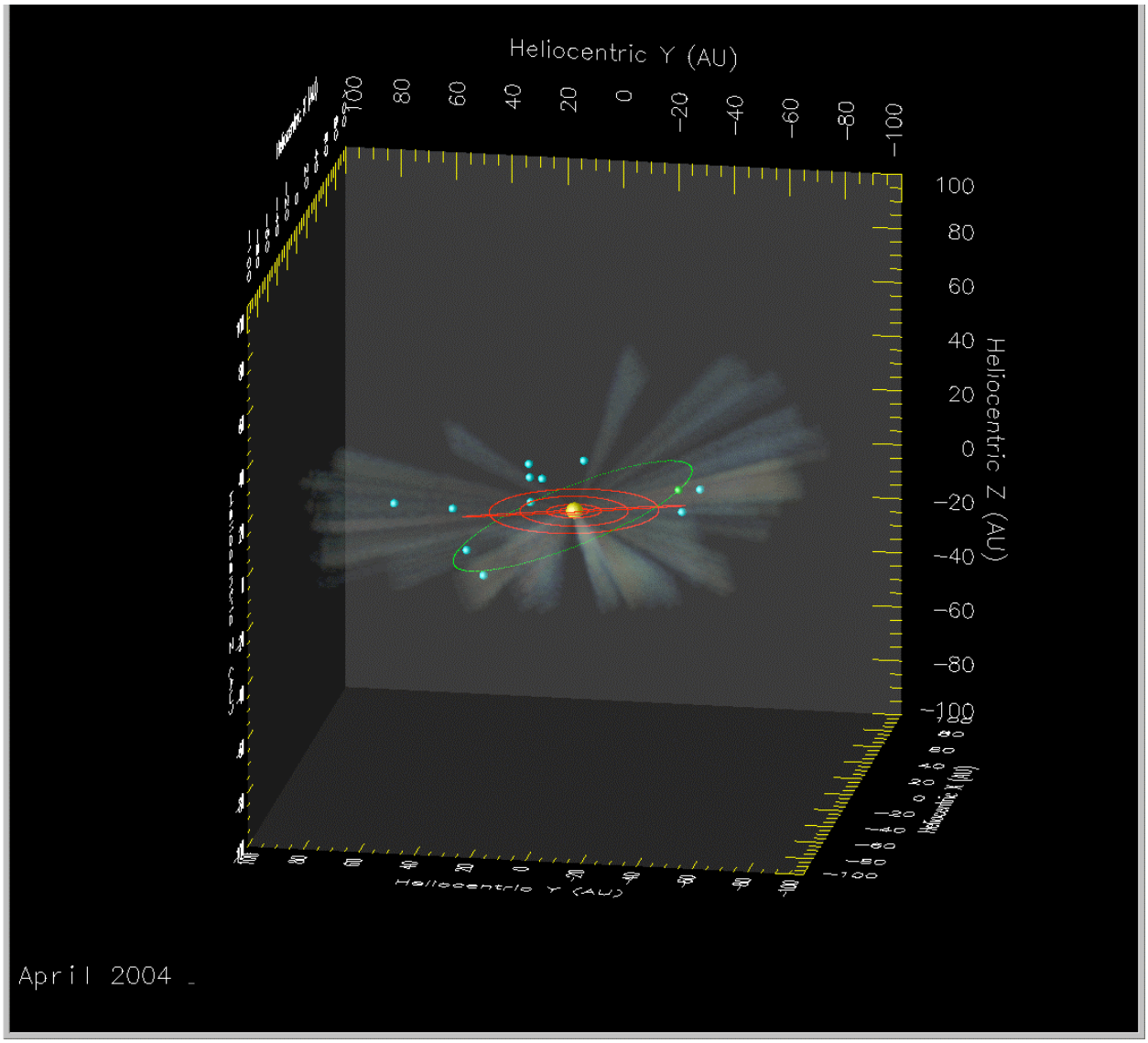

Figure 29: The survey coverage as of April 2004, one year after Figure 28. The additional reviews completed during that time have been added as additional lines of sight into the array. Note that, in addition to moving with a net counterclockwise motion, the original lines of sight from Figure 28 have also become less distinct, as their generated populations have dispersed over the year's timespan.

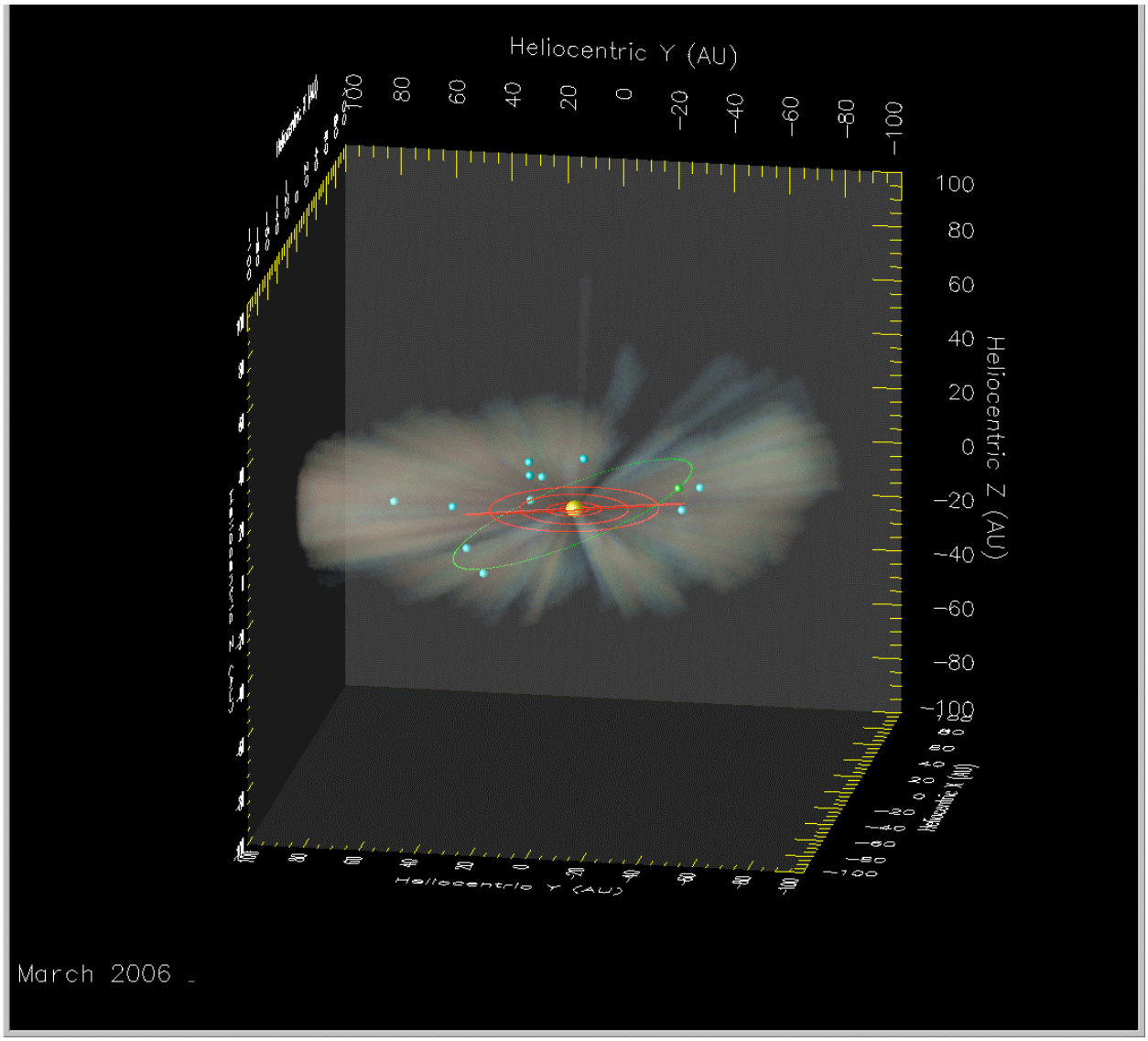

Figure 30: The final survey coverage as of March 2006. The overlap of the lines of sight acquired through the survey has created regions which were more heavily covered than others. This is represented by the shading of the array, with red representing regions of 100% coverage.

Figure 31 shows another plot of the final survey coverage, looking from the top down on the plane of the solar system. In this plot, the regions which were not covered, or only very sparsely, are readily apparent, with the largest region representing the intersection of the galactic plane and the ecliptic, looking towards the galactic center. As the time approaches in the year where

the ecliptic passes through the galactic plane and observations become impossible in that region, observers attempt to extend their data by shooting regions of ever higher of inclination on the edge of the galactic plane, essentially stepping up the plane away from the ecliptic. It is interesting to note that it was in this region, fairly thoroughly covered by numerous groups who practice this procedure, that our discovery and multiple re-detections of 2003 MW12 were made.

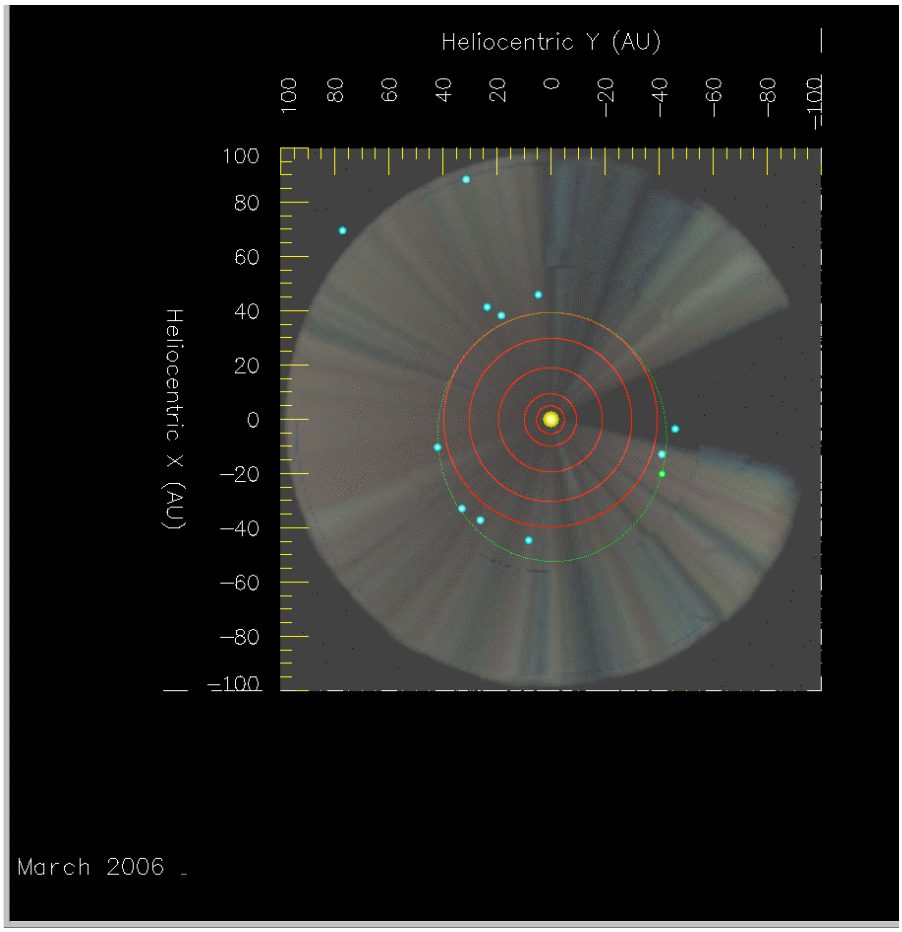

Figure 31: The final survey coverage when data collection ended in March 2006. The region of no coverage towards the galactic plane is shown as the empty region extending along the negative heliocentric y axis at x=0, and the region of lower coverage due to Kitt Peak's monsoon season is seen along the positive heliocentric x axis, at y=0. The region heavily covered by many observational groups, as they try to avoid the galactic center, is located along the bottom edge of the empty region looking towards the galactic center.

With the conclusion of the line of sight model over the data from March 2003 to April 2006, the biased total space coverage was calculated to be approximately 50% using this method.

2. Second Approach: Determination of the Probability a Candidate would be Detected

The second approach to determine the biased sky coverage of the survey was less rigorous, though still rooted in computer simulation. For this approach, a large randomized population of 10,602,900 objects was generated with orbital properties such as semi-major axis, inclination, eccentricity, and absolute magnitude within the range of those suggested for Planet X, and with randomly generated orbital angles, such as the Mean Anomaly, the Argument of Perihelion, and the Longitude of the Ascending Node, which served to orient the object's orbit in phase space with reference to the intersection of the galactic and ecliptic planes (called the Vernal Equinox). These objects were then placed in positions along their orbits, and the survey was rerun to determine how many of the objects would be detected. The number of objects detected versus the entire population would measure the survey coverage, and would also yield the efficiencies to which each orbital property was detected.

Using this large population, the survey was rerun using an automated program called Searchmosaic, written by Jim Scotti of the Spacewatch group, which given a time and a set of sky coordinates, would search the generated population and return the observational parameters, such as RA, Dec., RA rate, and Apparent Magnitude (*m*). The second program used, entitled Survey_Sim and written by Prof. Jeffrey Larsen of the Naval Academy, took the pointing history of the survey, the record of the locations and times of all images taken, and ran Searchmosaic while chronologically stepping through the survey. At each step, the coverage for

the survey was compared with the observing properties of any objects at the congruent location (given by the RA and Dec. of each object). Objects were marked as detected if their RA rate and Apparent Magnitudes were within the motion filters applied to SLOSUR, also taking into account SLOSUR's predetermined efficiency. At the end of the survey, the number of objects detected versus the total number of objects in the population yields an empirical calculation of the biased coverage of the survey. As 57% of the objects were detected, including efficiencies and filters, the biased coverage using this method was also 57%.

Moreover, because the objects detected had varying orbital properties, their detection rate would allow for the creation of histograms showing SLOSUR's efficiency at detecting each orbital property. These histograms are shown in Figures 32 and 33.

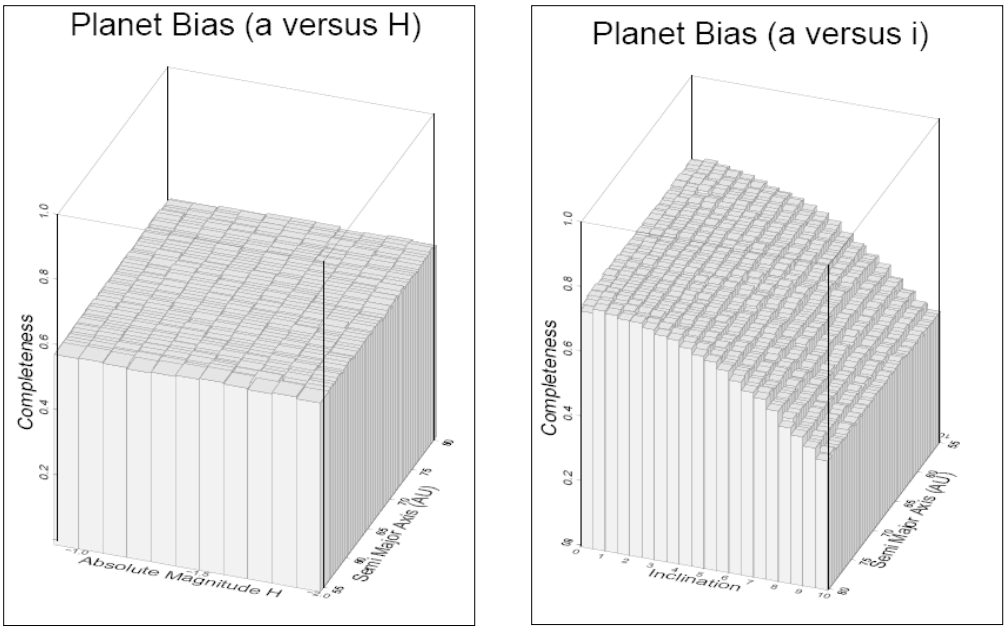

Figure 32: These histograms represent the percentage of objects detected as a function of absolute magnitude (H) versus semi-major axis (left) and orbital inclination versus semi-major axis (right). The null gradient of the left histogram shows that the survey was not affected by changes in the absolute magnitude of an object over the range of distances from 55 to 80 AU. The right histogram shows that the probability of detection decreases with a corresponding increase in orbital inclination. However, this probability was also independent of semi-major axis, as the same number of highly inclined objects found at 55 AU were found at 80 AU. This makes sense, as the data collection for the survey was tightly concentrated around the ecliptic, improving its odds of detecting an object of lower inclination.

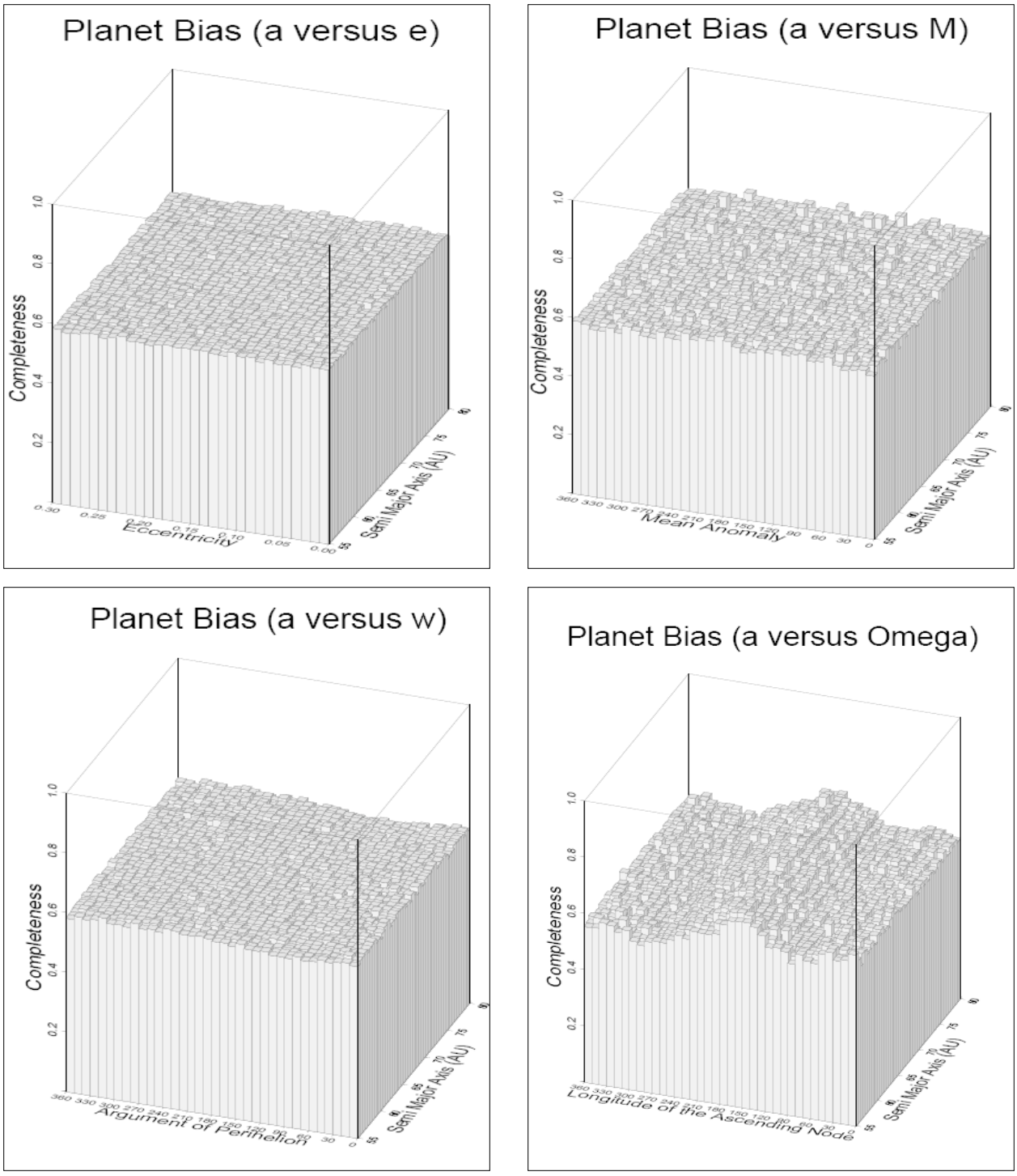

Figure 33: These histograms represent the percentage of objects detected as a function of semi-major axis versus orbital eccentricity (top left), semi-major axis versus the Mean Anomaly (top right), semi-major axis versus the Argument of Perihelion (bottom left), and semi-major axis versus the Longitude of the Ascending Node (bottom right). They show that the probability of detection was not affected by either orbital eccentricity, or variations in orientation from the Argument of Perihelion or the Mean Anomaly over a range of distances from 55 to 80 AU. The Longitude of the Ascending Node did affect the probability, however, for reasons that remain unexplained.

After variation of these properties over a range of semi-major axes from 55 to 80 AU, it was determined that the only properties influencing the probability of detection were the orbital inclination and the Longitude of the Ascending node (an angle in the solar plane defining the orientation of the point where an inclined orbit passes through the plane from south to north with reference to the Vernal Equinox). The consistency of detection found in the histograms of orbital eccentricity, absolute magnitude, Mean Anomaly, and Argument of Perihelion gives a strong argument for the coverage of the survey, as in each of these histograms, 57% of the objects were detected. Without influencing the survey by varying their parameters, then, the number of objects found with these properties must equal the biased volume of space searched by the survey.

Although the effect stemming from the Longitude of the Ascending Node remains unexplained, it is probable that its cause stems from the fact that the survey covered some regions to a higher extent than others. As the regions of sparse coverage (such as the galactic center and monsoon season regions) represent specific periods during the year, objects with Ascending Node Longitudes allowing them to be observable during this time were naturally biased against in the survey. The influence of the inclination on the probability of detection is purely a result of volume of space covered by the survey. As the survey was tightly concentrated on the ecliptic, at an inclination of zero degrees, objects spending large fractions of their orbits close to the ecliptic (low inclination orbits) were naturally more likely to be detected.

Chapter 6: Conclusions

This survey discovered a significant Kuiper Belt Object and detected 7 large previously known Kuiper Belt Objects and 1 Centaur (an object in orbit between Jupiter and Neptune), covering approximately 55% of the volume of space bounded by Planet X's proposed orbital properties (50% by volume, 57% by statistical detection). As shown by Figure 34 and 35, this figure increases to greater than 90% coverage if regions looking toward the galactic center, the monsoon season in Arizona, and inclinations of greater than 4 degrees are excluded.

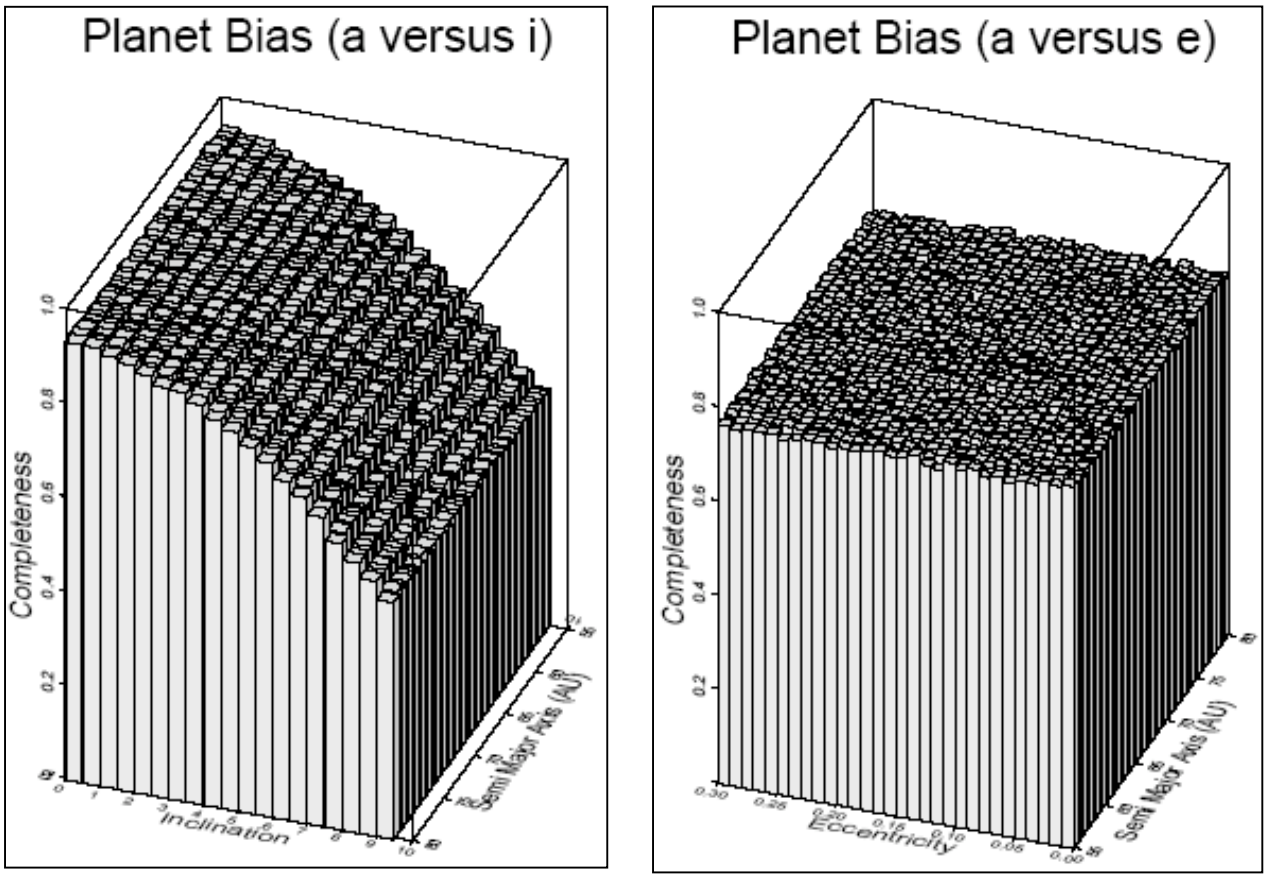

Figure 34: Histograms that represent the percentage of objects detected as a function of semi-major axis versus orbital inclination (left), and semi-major axis versus orbital eccentricity (right) for which the galactic center, monsoon season, and high inclination regions have been excluded. It is still apparent that the orbital inclination, while the eccentricity does not. however, the exclusion of these fields has increased the probability of detection.

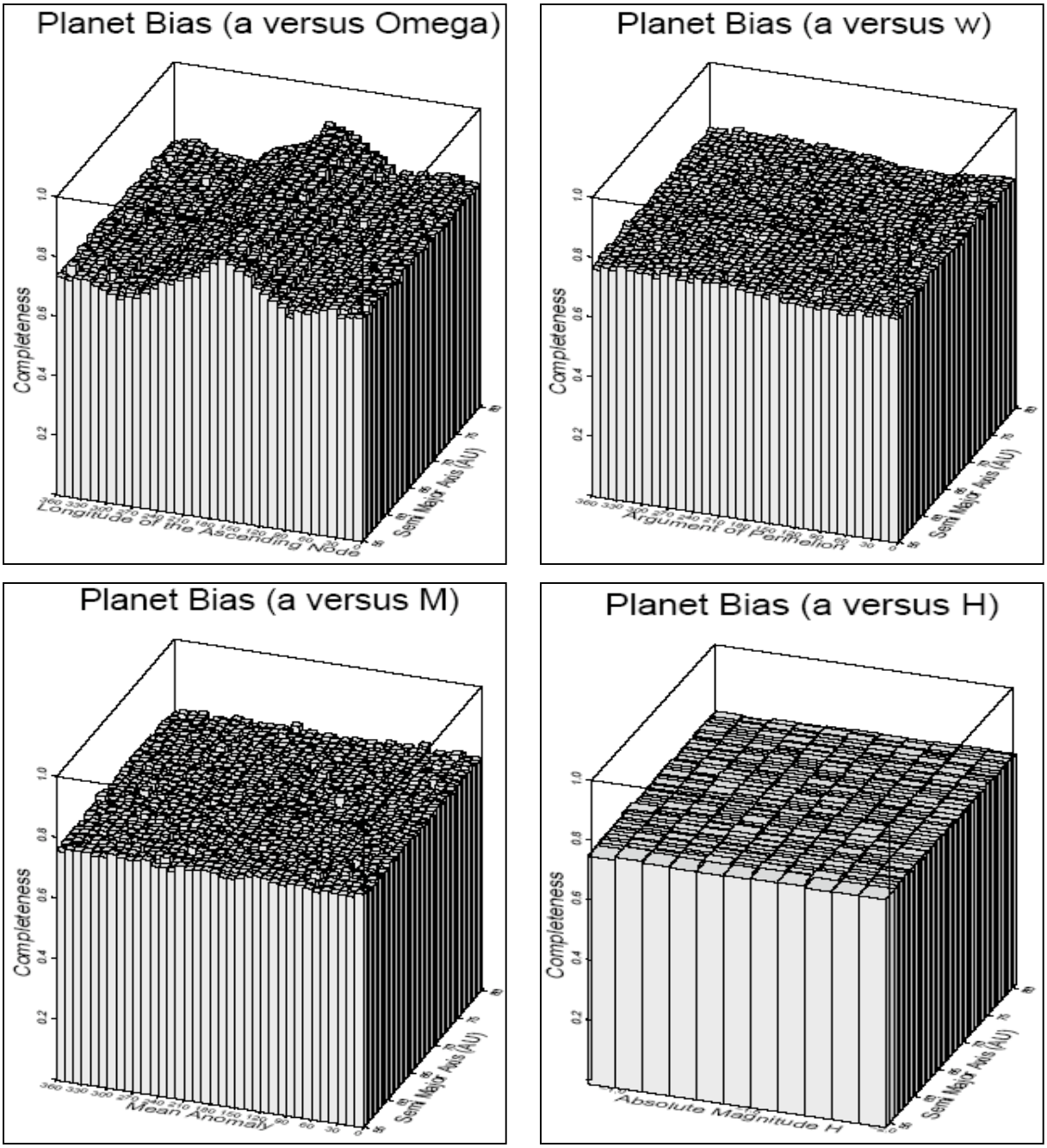

Figure 35: Histograms that represent the percentage of objects detected as a function of semi-major axis versus Longitude of the Ascending Node (top left), semi-major axis versus Argument of Perihelion (top right), semi-major axis versus Mean Anomaly (bottom left), and semi-major axis versus absolute magnitude (bottom right) for which the galactic center, monsoon season, and high inclination regions have been excluded. It is still apparent that the orbital inclination and the Longitude of the Ascending Node influence the probability of detection, with the interesting affect that removing the sparsely covered regions from the survey has further emphasized the unexplained structure of the efficiency histogram for the Longitude of the Ascending node.

More interesting, as 2003 MW12 was discovered in a region of the sky that has historically been highly searched by other astronomical survey groups, it can be argued that there are still numbers of large objects that may remain undiscovered in the Kuiper Belt. Alan Stern, a notable astronomer at the Southwest Research Institute and the leader of NASA's New Horizons Mission to Pluto, has predicted the existence of multiple significantly sized objects in the Kuiper Belt, based on computer models of the outer solar system.[18] However, these objects must reside either at inclinations greater than 4 degrees or within the sparsely covered galactic center and monsoon season. Further surveys should concentrate their coverage on these areas, as a result. The most profictable regions to be searched are near the ecliptic from 17-20 hours of RA (the galactic center), and 22-24 hours of RA (region of Arizona monsoon season).

Because a signal to noise ratio was included for a majority of the survey that was lower than necessary (for the detection of Planet X), the sensitivity of the search was increased to objects of lower magnitudes and greater distances from Earth. It is possible to place some constraints on whether the other side of the Kuiper Cliff should be visible from this information with a few simple assumptions. Figure 36 shows opposition distance versus visual apparent magnitude for several Kuiper Belt object sizes from 2003 MW12 (H=3.8 or roughly 4) up to the size of Mars (H=-2).

[18] Stern, S. Alan. "Regarding the Accretion of 2003 VB (Sedna) and Like Bodies in Distant Heliocentric Orbits."

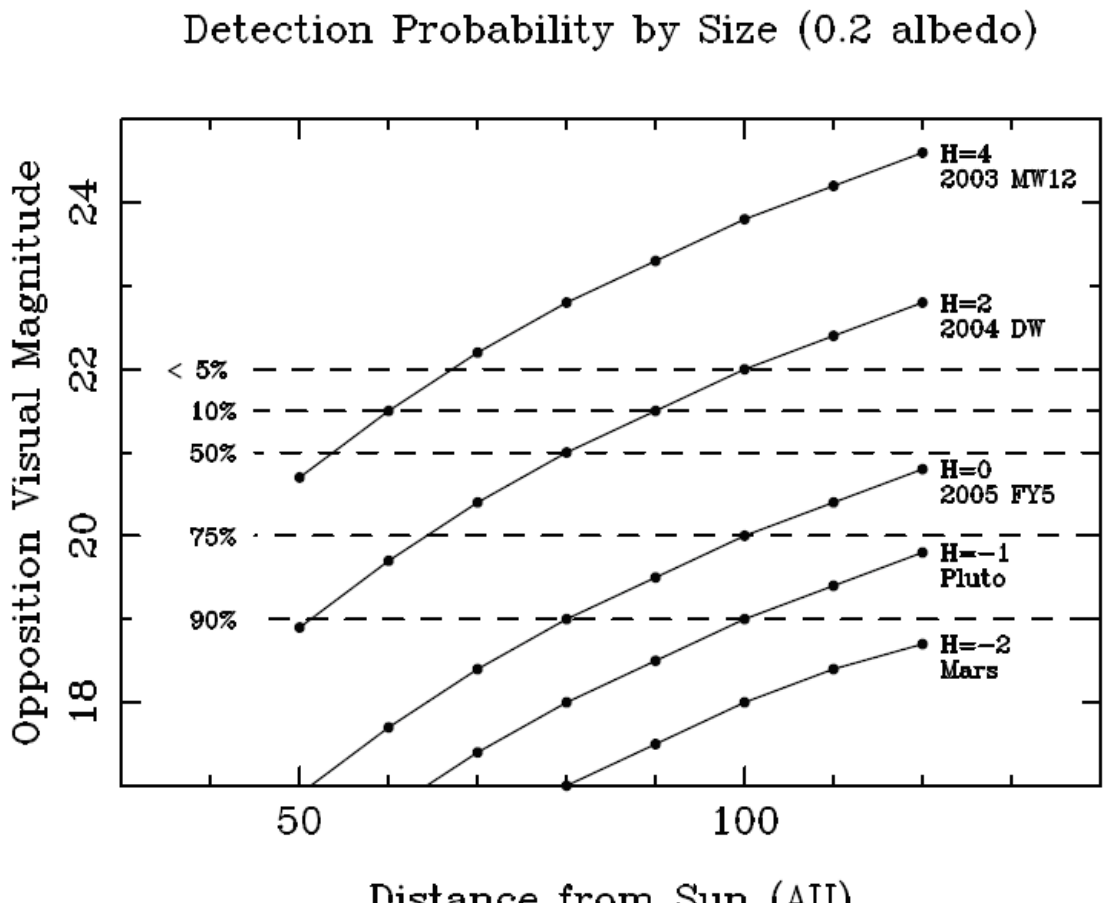

Figure 36: A plot of opposition distance versus visual apparent magnitude (m), giving the probability of detection by absolute magnitude (*H*), which can be related to size.

The curves are generated by the relation between asteroidal apparent (*m*) and absolute (*H*) magnitudes as well as distance from the Earth (Δ) and Sun (*r*) shown by Eq. 26.

$$m - H = 5\log(r\Delta) - 5 \qquad \textbf{Eq. 26}$$

Superimposed on the plot is our net survey efficiency of finding objects as a function of magnitude as determined in Chapter 4.

From this figure it is apparent that the survey had reasonable probabilities of detecting objects as large as H=0 out to 100 AU and a chance of detecting objects as small of H=2 over the same distance over more than half of the 10 degree wide ecliptic band. Under the assumption that the hypothetical outer band of the dynamically cold Kuiper Belt had just as many objects as

currently known in the inner band and all were 100 AU away (the outer limit of the planned sensitivity of this survey), it would lead to an estimated eight objects being detectable. Even in the worst case (all objects being 100 AU out) statistically three should have been found. If the band were closer than 100 AU, more objects should have been expected. Using the number of objects detected for each absolute magnitude, and the percentage of objects in the generated population from bias method 1 detected in this magnitude range (called the bias factor), the number of objects within this absolute magnitude range expected to exist can be estimated by dividing the number of detected objects by the bias factor. For magnitude ranges for which this survey did not detect any objects, the bias factor was used to place an upper bound on the number of objects estimated to exist. The plot showing the expected number of existing objects for a range of magnitudes from H=-2 to H=7 is shown in Table 4.

Absolute Magnitude (H) Range	Number Detected	Bias Factor	Estimated Number of Kuiper Belt Objects	Number of Kuiper Belt Objects Known
-2..-1	0	0.252	<4	0
-1..0	0	0.252	<4	1
0..1	0	0.252	<4	1
1..2	0	0.252	<4	0
2..3	0	0.214	<5	2
3..4	2	0.147	13	8
4..5	1	0.028	35	15
5..6	4	0.010	400	63
6..7	1	0.001	1000	170

Table 4: A table of the number of Kuiper Belt Objects expected to exist for each 1 magnitude range from H=-2 to H=7, calculated from the percentrage of these objects found from bias method 1 (bias factor) and the number detected in the survey for each magnitude range. Also shown are the number of Kuiper Belt Objects already discovered for each magnitude range. The number of known Kuiper Belt Objects was obtained from http://cfa-www.harvard.edu/iau/lists/TNOs.html.

Given the lack of detections of new objects past 80 AU, where the end to the resonance gap caused by the Kuiper Cliff should be located, the probability of finding objects at such

distances, and the efficiency of this survey, a fairly firm conclusion can be drawn that while this survey does see the Kuiper Cliff it does not see the gap ending anywhere. Two possibilities explain this result. First, either the other side of the resonance gap is located farther away than the 75 AU proposed by Brunini and Melita's theories and past the 100 AU detection limit of this survey or it does not exist and the gap extends indefinitely. This is difficult for current dynamical theories to explain, due largely to the fact that the Kuiper Cliff is so sharply truncated.[19] However, it may be a natural consequence of star formation and conditions in the early solar nebula.[20] Second, the size distribution may not be constant on the other side of the gap as proposed for the EKBO's.[21] This scenario is plausible if densities in the outer solar system decrease as one moves away from the Sun more rapidly for the outer belt than the known belt, and so there are no large objects for this survey to detect. Current theories about this are in flux however. While Bernstein et al. (2004) indicate that large objects are rare, Stern (2005) has hypothesized that a period of rapid accretion early in the history of the Solar System may well be able to form many larger bodies.[22] This survey's detections in either case have the same consequences as the observational limits set by Allen, Bernstein and Malhotra (2002) but extends the distance considerably from 60 AU to 100 AU for large objects.[23]

This study, after searching a majority of "dynamically cold" (low inclination) Classical Kuiper Belt, did not provide any evidence of larger objects at large distances (> 55 AU) which could represent Planet X. Given the existence of higher inclination large bodies, however, one

---

[19] Allen, R.L., G.M. Bernstein and R. Malhotra, "Observational Limits on a Distant Cold Kuiper Belt," in *The Astronomical Journal*, Vol. 124 (2002): pp. 2949-2954.
[20] Vicente, S.M., and J. Alves, "Size Distribution of Circumstellar Disks in the Trapezium Cluster," in *Astronomy & Astrophysics*, Vol. 441 (2005): pp. 195-205.
[21] Bernstein, G.M. et al., "The Size Distribution of Trans-Neptunian Bodies," in *The Astronomical Journal*, Vol. 128 (2004): pp. 1364-1390.
[22] Stern, S. Alan. "Regarding the Accretion of 2003 VB (Sedna) and Like Bodies in Distant Heliocentric Orbits."
[23] Allen, R.L., G.M. Bernstein and R. Malhotra, "Observational Limits on a Distant Cold Kuiper Belt,"

of the remaining challenges is to explain how high inclination objects with circular orbits (such as 2004 XR_{190}[24]) can be created while the inner Kuiper Belt is so sharply truncated past the 2:1 Neptune resonance. Being outside the scope of this study both in observable sky and theory (as it was focused on such low inclinations), this study's analysis was independent of these Scattered Disk objects (SEKBO's), which may themselves form a link to the inner Oort Cloud.[25] Given the number of recent detections of large Scattered Disk objects such as 2003 UB_{313}, 2005 $FY_{9,}$, and 2003 EL_{61}, the apparent lack of similarly sized objects in the Classical region, and the recent discovery that most proto planetary disks are well over 100 AU in diameter, the next few years should be crucial to understanding the history and dynamics of the outer solar system.[26]

[24] Allen, R.L. et al.
[25] Brown, Michael E., Chadwick Trujillo, and David Rabinowitz, "Discovery of a Candidate Inner Oort Cloud Planetoid," in *The Astrophysical Journal*, Vol. 617 (2004): pp. 645-649.
[26] Vicente, S.M., and J. Alves.

Bibliography

Allen, R.L. et al. “Discover of a Low-Eccentricity, High-Inclination Kuiper Belt Object at 48 AU.” In *The Astrophysical Journal*, Vol. 640 (2006): pp. L83-L86.

Allen, R.L., G.M. Bernstein and R. Malhotra. “Observational Limits on a Distant Cold Kuiper Belt.” In *The Astronomical Journal*, Vol. 124 (2002): pp. 2949-2954.

Beals, Gerald. “Thomas Edison Quotes.” <http://www.thomasedison.com/edquote.htm>, (1996): 7 December, 2004.

Bernstein, G.M. et al. “The Size Distribution of Trans-Neptunian Bodies.” In *The Astronomical Journal*, Vol. 128 (2004): pp. 1364-1390.

Bernstein, Gary and Bharat Khushalani. “Orbit Fitting and Uncertainties for Kuiper Belt Objects.” In *The Astronomical Journal*, Vol. 120 (2000): pp. 3323-3332.

Brown, Michael E., Chadwick Trujillo, and David Rabinowitz. “Discovery of a Candidate Inner Oort Cloud Planetoid.” In *The Astrophysical Journal*, Vol. 617 (2004): pp. 645-649.

Brown, M.E., C.A. Trujillo, and D. L. Rabinowitz, “Discovery of a Planetary-Sized Object in the Scattered Kuiper Belt,” in *The Astrophysical Journal*, Vol. 635 (2005): pp. L97-L100.

Brunini, Adrian and Julio A. Fernandez. “Perturbations on an Extended Kuiper Disk Caused by Passing Stars and Giant Molecular Clouds.” In *Astronomy and Astrophysics*, Vol. 308 (1996): pp. 988-994.

Brunini, Adrian and M.D. Melita. “The Existence of a Planet beyond 50 AU and the Orbital Distribution of the Classical Edgeworth-Kuiper-Belt Objects.” In *Icarus*, Vol. 160 (2002): pp. 32-43.

Bruton, Dan. “Conversion of Absolute Magnitude to Diameter for Minor Planets.” <http://www.physics.sfasu.edu/astro/asteroids/sizemagnitude.html>, 1 February 2006.

Chambers, J.E. “A Hybrid Symplectic Integrator that Permits Close Encounters between Massive Bodies.” In *Monthly Notices of the Royal Astronomical Society*, Vol. 000 (1998): pp. 1-8.

Couper, Heather and Nigel Henbest. “The Hunt for Planet X.” In *New Scientist*, Vol. 176 (2002): pp. 30-34.

De la Fuente Marcos, C. and R. de la Fuente Marcos. “Reshaping the Outskirts of Planetary Systems.” In *Astronomy and Astrophysics*, Vol. 371 (2001): pp. 1097-1106.

Dermott, Stanley F. and Carl D. Murray. *Solar System Dynamics*. United States: Cambridge University Press, 1999.

Fernandez, Julio A. and W.H. Ip. "Orbital Expansion and Resonant Trapping During the Late Accretion Stages of the Outer Planets." In *Planetary Space Science*, Vol. 44 No. 5 (1996): pp. 431-439.

Freedman, Roger A. and William J. Kaufmann III. *Universe*. 7th ed. New York: W.H. Freeman and Company, 2005.

Hamilton, Calvin J. "The Kuiper Belt." Views of the Solar System. <http://www.solarviews.com/eng/kuiper.htm>, (2001): 8 December 2004.

Ida, Shigeru, John Larwood, and Andreas Burkert. "Evidence of Early Stellar Encounters in the Orbital Distribution of Edgeworth-Kuiper Belt Objects." In *The Astrophysical Journal*, Vol. 528 (2000): pp. 351-356.

Jewitt, David, Jane Luu and Chadwick Trujillo. "Large Kuiper Belt Objects: The Mauna Kea 8K CCD Survey." in *Astronomical Journal*, Vol. 115 (1998): pp. 2125-2135.

Jewitt, D.C. and J.X. Luu, "Discover of the Candidate Kuiper Belt Object 1992 QB_1," in *Nature*, Vol. 362 (1992): pp. 730-732.

"Kirkwood Gap." In the Fablis Online Dictionary. <http://encyclopedia.fablis.com/index.php/Kirkwood_gap> (2004): 15 January 2005.

Larsen, Jeffrey A. et al. "The Spacewatch Wide-Area Survey for Bright Centaurs and Trans-Neptunian Objects." In *The Astronomical Journal*, Vol. 121 (2001): pp. 562-579.

Malhotra, Renu. "The Origin of Pluto's Orbit: Implications for the Solar System." In *The Astronomical Journal*, Vol. 110 no. 1 (1995): pp. 420-429.

"Most Distant Object in Solar System Discovered." Jet Propulsion Laboratory, <http://www.jpl.nasa.gov/releases/2004/85.cfm>, (2004): 5 January 2005.

"The Spacewatch Project." < http://spacewatch.lpl.arizona.edu/> (31 October 2003): 15 January 2005.

Stern, S. Alan. "Regarding the Accretion of 2003 VB (Sedna) and Like Bodies in Distant Heliocentric Orbits." In *The Astronomical Journal*, Vol. 129 (2005): pp. 526-529.

Taylor, John R. *Classical Mechanics*. United States: University Science Books, 2005.

Trujillo, C.A., D.C. Jewitt and J.X. Luu. “Properties of the trans-neptunian belt: Statistics of the CFHT survey.” In *The Astronomical Journal*, Vol. 122 (2001): pp. 157-473.

Tytell, David. “The New Kings of the Kuiper Belt.” In *Sky and Telescope*, October 2005, p. 31.

Vicente, S.M., and J. Alves. “Size Distribution of Circumstellar Disks in the Trapezium Cluster.” In *Astronomy & Astrophysics*, Vol. 441 (2005): pp. 195-205.

Appendix A: Glossary

Absolute Magnitude: The brightness of an object, as seen from Earth, if placed at a standardized distance from the Earth.

Albedo: The percentage of incident light reflecting off an object.

Apparent Magnitude: The brightness of an object, as seen from Earth, at the object's actual distance from the Earth.

AU: a unit of distance, equal to the distance between the Sun and the Earth.

CCD: Charge-coupled-device. A standard astronomical camera using a light sensing computer chip to digitally record images of the sky.

CEKBO: Classical Edgeworth-Kuiper Belt Object. A classification for asteroids in the Kuiper Belt that have low eccentricity orbits, and are generally found from 39 to 47 AU.

Eccentricity: The ratio of focal distance to the semi-major axis. This term describes how circular an orbit is, with an eccentricity of 0 equal to a perfect circle and an eccentricity of 1 being a parabola.

Ecliptic: The plane of the solar system, defined as the plane passing through the Sun and Earth's orbit.

Inclination: The angle measuring how tilted an orbit is relative to the ecliptic.

Kuiper Cliff: A term for the abrupt absence of CEKBO's in the Kuiper Belt, past 47 AU from the sun.

Plutino: A classification for asteroids in the Kuiper Belt with orbital properties similar to Pluto. This group is located at 39 AU, and is in a 2:3 mean-motion resonance with Neptune.

Resonance: An integer ratio of any two periods of motion. This is commonly seen in the solar system as a ratio between two periods of rotation or revolution.

SEKBO: Scattered Edgeworth-Kuiper Belt Object. A classification for asteroids in the Kuiper Belt that have high eccentricity, high inclination orbits.

Semi-major Axis: Half the distance of the major leg of an object's orbit.

SLOSUR: Computer software used to analyze raw images and identify moving objects in that image.

Appendix B: Bias Program Code

Platform Technical Information:

Apple PowerMac G4 laptop
1.67 GHz processor
1 GB RAM
Mac OS X 10.4 “Tiger”

Program run on XII terminal, based on Darwin version of Unix, runs on Perl version 5.8.6 and uses the external programs:
(1) Time::JulianDay
Created by David Muir Sharnoff
(2) Ephem
Created by Jim Scotti, Spacewatch Project and Davis Tholen, Univ. of Hawaii

```
##############################################
#                                            #
#          Bias Computer Model Program       #
#           Created by Eric Roe              #
#           Version 3.0                      #
##############################################


#!/usr/bin/perl


use Math::Trig;
#use "/Users/casparwhittey/Trident/slalib_c/slalib.h";
#use "/Users/casparwhittey/Trident/slali_c/slamac.h";
use Time::JulianDay;

$jd = julian_day(2005, 10, 28);
print "$jd\n";
$mjd = $jd - 2400000;

$obj = 10000;

##############################################
#Creating a population of $obj random objects#
##############################################
#Ranges:  a=55..75AU, i=0..10,
$H = 18.60;
`rm ./myelement.dat`;
printf "Generating KBO population:\n";
open OF, ">KBOelement";

for ($i=1; $i<$obj; $i++) {
   $nmbr = $i;
   sprintf ("%02f",$nmbr);
   if ($nmbr<10000) {
              if ($nmbr<1000) {
                 if ($nmbr<100) {
                           if ($nmbr<10) {
                              printf OF "     $nmbr 2005 XX         ";
                           }
                           else {
                              printf OF "    $nmbr 2005 XX         ";
                           }
                 }
```

```
            else {
                printf OF "   $nmbr 2005 XX         ";
            }
        }
        else {
            printf OF "  $nmbr 2005 XX         ";
        }
}

$a = rand(20)+55;
$e = rand(1)*0.3;
$incl = rand(10);
$angle1 = rand(360);
$angle2 = rand(360);
$angle3 = rand(360);

$a = sprintf("%.7f",$a);
$e = sprintf("%.7f",$e);
$e = substr($e,1,100);
$incl = sprintf("%.5f",$incl);
$angle1 = sprintf("%.5f",$angle1);
$angle2 = sprintf("%.5f",$angle2);
$angle3 = sprintf("%.5f",$angle3);

printf OF "$a $e";

if ($incl<10) {
        printf OF "  $incl";
}
else {
        printf OF " $incl";
}

if ($angle1<100) {
        if ($angle1<10) {
            printf OF "   $angle1";
        }
        else {
            printf OF "  $angle1";
        }
}
else {
        printf OF " $angle1";
}

if ($angle2<100) {
    if ($angle2<10) {
        printf OF "   $angle2";
    }
    else {
        printf OF "  $angle2";
    }
}
else {
    printf OF " $angle2";
}

if ($angle3<100) {
    if ($angle3<10) {
        printf OF "   $angle3";
    }
    else {
        printf OF "  $angle3";
    }
}
else {
    printf OF " $angle3";
}
```

```
   printf OF " $mjd";
   printf OF " $H";
   $lastline = "   .15    0      KBO  173d MPO 70208     \n";
   printf OF "$lastline";
   chop ($lastline);


}

`cp ./KBOelement /sw/data/astr/myelement.dat`;
`cp ./KBOelement ./myelement.dat`;
#`rm ./KBOelement`;

printf "KBOelement file creation done. Results found in /Users/casparwhittey/Trident/programming/bias/KBOelement\n";
printf "KBOelement file copied to /sw/data/astr/myelement.dat\n";

##############################
#GETTING ORBITS FOR 1ST MONTH#
##############################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month = 1;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2003";   #max ending day is 99 days from start day#
$endmonth = "06";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
            $days = 31;
            printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
```

```
#`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 1st month#
##############################
$num = 1;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {

            ###Coords are (x0=R0,y0=0,z0=0) ######

            $j++;

            if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
               $R0 = substr($line,45,8);
               $x0 = substr($line,80,8);
               $y0 = substr($line,89,8);
               $z0 = substr($line,98,8);

               $a0 = 0;

               $R0 = sprintf("%8.4f",$R0);

               $lastline = "$R0 $a0 $a0 0.00\n";
               printf OF1 "$lastline";
               chop ($lastline);
            }

                  ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

            if ($j == 7) {                    ###final position, check ephem.m file to check line position###
               $R = substr($line,45,8);
               $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

               if ($x!=0) {
                        $theta = atan($y/$x);
                        $theta0 = atan($y0/$x0);
                        $delta_theta = $theta - $theta0;

                        $delta_y = $R*sin($delta_theta);
                        $delta_y = abs($delta_y);
                        $delta_z = $z - $z0;

                        $R = sprintf("%8.4f",$R);
                        $delta_y = sprintf("%8.4f",$delta_y);
                        $delta_z = sprintf("%8.4f",$delta_z);

#                  if ($delta_y>0) {
                        $lastline = "$R $delta_y $delta_z 0.00\n";
                        printf OF2 "$lastline";
                        chop ($lastline);
```

```
#                   }
                    }
                }
    }
    close IF;
}
printf "initial (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`cp ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#The ephem output has this justification
#0        1        2        3        4        5        6        7        8        9        0        1        2
#012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789
#
# 2005 Nov  1  1:00c 23 00 50.84 -00 03 45.9  43.0552  42.3168 355.5806   1.8457  42.9520  -2.6490   1.3629
# 2005 Dec  1  1:00n 23 00 52.72 -00 11 35.2  43.0091  42.7081 355.3065   1.8220  42.9103  -2.5776   1.3578
#
# Year Mon DY  Time  RH RM RSSSS DDD DM DSSS   RSun     Delta  LONGITUD LATITUDE   X(AU)    Y(AU)    Z(AU)
#
#
#$magnitude = substr($line,70,6);

#######################################
#Creation of ephem file for 2nd month#
#######################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2003";   #max ending day is 99 days from start day#
$endmonth = "07";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
```

```
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 2nd month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
            if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 7) {                  ###final position, check ephem.m file to check line positionn###
               $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

               if ($x!=0) {
                     $theta = atan($y/$x);
                     $theta0 = atan($y0/$x0);
                     $delta_theta = $theta - $theta0;

                     $delta_y = $R*sin($delta_theta);
                     $delta_y = abs($delta_y);
                     $delta_z = $z - $z0;
```

```
                        $R = sprintf("%8.4f",$R);
                        $delta_y = sprintf("%8.4f",$delta_y);
                        $delta_z = sprintf("%8.4f",$delta_z);

        if ($delta_y>0) {
           $lastline = "$R $delta_y $delta_z 0.00\n";
           printf OF2 "$lastline";
           chop ($lastline);
        }
              }
            }
   }
   close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 3rd month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2003";   #max ending day is 99 days from start day#
$endmonth = "08";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
```

```
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 3rd month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;


open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######
            $j++;

            if ($j == 6) {                  ##initial position, check ephem.m file to check line position###
        $R0 = substr($line,45,8);
        $x0 = substr($line,80,8);
        $y0 = substr($line,89,8);
        $z0 = substr($line,98,8);

        $a0 = 0;

        $R0 = sprintf("%8.4f",$R0);

        $lastline = "$R0 $a0 $a0 0.00\n";
        printf OF1 "$lastline";
        chop ($lastline);
    }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

    if ($j == 7) {                  ###final position, check ephem.m file to check line position###
        $R = substr($line,45,8);
        $x = substr($line,80,8);
        $y = substr($line,89,8);
        $z = substr($line,98,8);

            if ($x!=0) {
                $theta = atan($y/$x);
                $theta0 = atan($y0/$x0);
                $delta_theta = $theta - $theta0;

                $delta_y = $R*sin($delta_theta);
                $delta_y = abs($delta_y);
                $delta_z = $z - $z0;

                $R = sprintf("%8.4f",$R);
                $delta_y = sprintf("%8.4f",$delta_y);
```

```
                        $delta_z = sprintf("%8.4f",$delta_z);

                        if ($delta_y>0) {
                           $lastline = "$R $delta_y $delta_z 0.00\n";
                           printf OF2 "$lastline";
                           chop ($lastline);
                        }
                     }
                  }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial3.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal3.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 4th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2003";   #max ending day is 99 days from start day#
$endmonth = "09";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";
```

```
    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 4th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

        $j++;
                if ($j == 6) {                  ##initial position, check ephem.m file to check line position###
            $R0 = substr($line,45,8);
            $x0 = substr($line,80,8);
            $y0 = substr($line,89,8);
            $z0 = substr($line,98,8);

            $a0 = 0;

            $R0 = sprintf("%8.4f",$R0);

            $lastline = "$R0 $a0 $a0 0.00\n";
            printf OF1 "$lastline";
            chop ($lastline);
        }

                ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

        if ($j == 10) {                 ###final position, check ephem.m file to check line position###
                    $R = substr($line,45,8);
            $x = substr($line,80,8);
            $y = substr($line,89,8);
            $z = substr($line,98,8);

                    if ($x!=0) {
                                $theta = atan($y/$x);
                                $theta0 = atan($y0/$x0);
                                $delta_theta = $theta - $theta0;

                                $delta_y = $R*sin($delta_theta);
                                $delta_y = abs($delta_y);
                                $delta_z = $z - $z0;

                                $R = sprintf("%8.4f",$R);
                                $delta_y = sprintf("%8.4f",$delta_y);
                                $delta_z = sprintf("%8.4f",$delta_z);

                                if ($delta_y>0) {
```

```
                        $lastline = "$R $delta_y $delta_z 0.00\n";
                        printf OF2 "$lastline";
                        chop ($lastline);
                    }
                }
            }
    }
    close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 5th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2003";   #max ending day is 99 days from start day#
$endmonth = "10";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
    $days = "0$days";
}
else {
    if ($days>99) {
        $days = 31;
        printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
    }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
    open OF, ">epheminputfile1";
    printf OF "i\n";
    printf OF "$styr/$stmonth/$stday1 01:00:00\n";
    printf OF "$endyr/$endmonth/$endday 01:00\n";
    printf OF "$period\n";
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
```

```
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 5th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

          if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
        $R0 = substr($line,45,8);
        $x0 = substr($line,80,8);
        $y0 = substr($line,89,8);
        $z0 = substr($line,98,8);

        $a0 = 0;

        $R0 = sprintf("%8.4f",$R0);

        $lastline = "$R0 $a0 $a0 0.00\n";
        printf OF1 "$lastline";
        chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


      if ($j == 11) {                   ###final position, check ephem.m file to check line position###
        $R = substr($line,45,8);
        $x = substr($line,80,8);
        $y = substr($line,89,8);
        $z = substr($line,98,8);

              if ($x!=0) {
                    $theta = atan($y/$x);
                    $theta0 = atan($y0/$x0);
                    $delta_theta = $theta - $theta0;

                    $delta_y = $R*sin($delta_theta);
                    $delta_y = abs($delta_y);
                    $delta_z = $z - $z0;

                    $R = sprintf("%8.4f",$R);
                    $delta_y = sprintf("%8.4f",$delta_y);
                    $delta_z = sprintf("%8.4f",$delta_z);

                    if ($delta_y>0) {
                       $lastline = "$R $delta_y $delta_z 0.00\n";
                       printf OF2 "$lastline";
```

```
                        chop ($lastline);
                    }
                }
            }
    }
    close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


####################################
#Creation of ephem file for 6th month#
####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2003";   #max ending day is 99 days from start day#
$endmonth = "11";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
    $days = "0$days";
}
else {
    if ($days>99) {
        $days = 31;
        printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
    }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
    open OF, ">epheminputfile1";
    printf OF "i\n";
    printf OF "$styr/$stmonth/$stday1 01:00:00\n";
    printf OF "$endyr/$endmonth/$endday 01:00\n";
    printf OF "$period\n";
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
```

```
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

####################################
#Extract coords for 6th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

          if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


      if ($j == 12) {                  ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

                     $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
```

```
                chop ($lastline);
            }
        }
    }
  }
  close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 7th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2003";   #max ending day is 99 days from start day#
$endmonth = "12";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
```

```
}
`rm ./epheminputfile1`;
printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

####################################
#Extract coords for 7th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


      if ($j == 13) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
```

```
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 8th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "01";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;
```

```
printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";


####################################
#Extract coords for 8th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


      if ($j == 14) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

               if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
```

```
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 9nd month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "02";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;
```

```
printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 9nd month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 15) {                   ###final position, check ephem.m file to check line positionn###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

                        if ($delta_y>0) {
                           $lastline = "$R $delta_y $delta_z 0.00\n";
                           printf OF2 "$lastline";
                           chop ($lastline);
                        }
         }
      }
```

```
   }
   close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 10th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "03";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
```

```
#Extract coords for 10th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;


open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######
      $j++;

      if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);
                $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 16) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);


            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}
```

```
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

####################################
#Creation of ephem file for 11th month#
####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "04";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 11th month#
##############################
```

```
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }
            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 17) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
```

```
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 12th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "05";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 12th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
```

```
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 18) {                   ###final position, check ephem.m file to check line positionn###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

                        if ($delta_y>0) {
                           $lastline = "$R $delta_y $delta_z 0.00\n";
                           printf OF2 "$lastline";
                           chop ($lastline);
                        }
         }
      }
   }
   close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
```

```
####################################
#Creation of ephem file for 13th month#
####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "06";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 13th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;
```

```
open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######
      $j++;

      if ($j == 6) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 19) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
```

```
#Creation of ephem file for 14th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "07";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 14th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";
```

```
foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

               $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 17) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 15th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
```

```
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "08";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 15th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;
printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
```

```
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
              }

              ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


      if ($j == 18) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 16th  month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
```

```
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "09";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 16th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
```

```
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 19) {                   ###final position, check ephem.m file to check line positionn###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

                        if ($delta_y>0) {
                           $lastline = "$R $delta_y $delta_z 0.00\n";
                           printf OF2 "$lastline";
                           chop ($lastline);
                        }
         }
      }
   }
   close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial2.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal2.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 17th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
```

```
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "10";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 17th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;


open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######
      $j++;
```

```
    if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
       $R0 = substr($line,45,8);
       $x0 = substr($line,80,8);
       $y0 = substr($line,89,8);
       $z0 = substr($line,98,8);

       $a0 = 0;

       $R0 = sprintf("%8.4f",$R0);

       $lastline = "$R0 $a0 $a0 0.00\n";
       printf OF1 "$lastline";
       chop ($lastline);
    }

          ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

    if ($j == 20) {                   ###final position, check ephem.m file to check line position###
       $R = substr($line,45,8);
       $x = substr($line,80,8);
       $y = substr($line,89,8);
       $z = substr($line,98,8);

       if ($x!=0) {
          $theta = atan($y/$x);
          $theta0 = atan($y0/$x0);
          $delta_theta = $theta - $theta0;

          $delta_y = $R*sin($delta_theta);
          $delta_y = abs($delta_y);
          $delta_z = $z - $z0;

          $R = sprintf("%8.4f",$R);
          $delta_y = sprintf("%8.4f",$delta_y);
          $delta_z = sprintf("%8.4f",$delta_z);

          if ($delta_y>0) {
             $lastline = "$R $delta_y $delta_z 0.00\n";
             printf OF2 "$lastline";
             chop ($lastline);
          }
       }
    }
  }
  close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 18th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#
```

```
$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "11";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 18th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
```

```
        $y0 = substr($line,89,8);
        $z0 = substr($line,98,8);

        $a0 = 0;

        $R0 = sprintf("%8.4f",$R0);

        $lastline = "$R0 $a0 $a0 0.00\n";
        printf OF1 "$lastline";
        chop ($lastline);
    }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

    if ($j == 21) {                 ###final position, check ephem.m file to check line position###
        $R = substr($line,45,8);
        $x = substr($line,80,8);
        $y = substr($line,89,8);
        $z = substr($line,98,8);

        if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
                $lastline = "$R $delta_y $delta_z 0.00\n";
                printf OF2 "$lastline";
                chop ($lastline);
            }
        }
    }
  }
  close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 19th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2004";   #max ending day is 99 days from start day#
$endmonth = "12";
$endday = "01";
```

```
$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 19th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);
```

```
        $a0 = 0;

        $R0 = sprintf("%8.4f",$R0);

        $lastline = "$R0 $a0 $a0 0.00\n";
        printf OF1 "$lastline";
        chop ($lastline);
    }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


    if ($j == 22) {                 ###final position, check ephem.m file to check line position###
        $R = substr($line,45,8);
        $x = substr($line,80,8);
        $y = substr($line,89,8);
        $z = substr($line,98,8);

        if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
                $lastline = "$R $delta_y $delta_z 0.00\n";
                printf OF2 "$lastline";
                chop ($lastline);
            }
        }
    }
  }
  close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 20th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "01";
$endday = "01";
```

```
$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

####################################
#Extract coords for 20th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);
```

```
        $a0 = 0;

        $R0 = sprintf("%8.4f",$R0);

        $lastline = "$R0 $a0 $a0 0.00\n";
        printf OF1 "$lastline";
        chop ($lastline);
    }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


    if ($j == 23) {                 ###final position, check ephem.m file to check line position###
        $R = substr($line,45,8);
        $x = substr($line,80,8);
        $y = substr($line,89,8);
        $z = substr($line,98,8);

        if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
                $lastline = "$R $delta_y $delta_z 0.00\n";
                printf OF2 "$lastline";
                chop ($lastline);
            }
        }
    }
  }
  close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 21st month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "02";
$endday = "01";
```

```
$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 21st month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;
```

```
        $R0 = sprintf("%8.4f",$R0);

                $lastline = "$R0 $a0 $a0 0.00\n";
        printf OF1 "$lastline";
        chop ($lastline);
    }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

    if ($j == 24) {                 ###final position, check ephem.m file to check line positionn###
        $R = substr($line,45,8);
        $x = substr($line,80,8);
        $y = substr($line,89,8);
        $z = substr($line,98,8);

        if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

                        if ($delta_y>0) {
                $lastline = "$R $delta_y $delta_z 0.00\n";
                printf OF2 "$lastline";
                chop ($lastline);
            }
        }
    }
  }
  close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 22nd month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "03";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
```

```
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 22nd month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;


open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######
      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);
```

```
        $lastline = "$R0 $a0 $a0 0.00\n";
        printf OF1 "$lastline";
        chop ($lastline);
    }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

    if ($j == 25) {                 ###final position, check ephem.m file to check line position###
        $R = substr($line,45,8);
        $x = substr($line,80,8);
        $y = substr($line,89,8);
        $z = substr($line,98,8);

        if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
                $lastline = "$R $delta_y $delta_z 0.00\n";
                printf OF2 "$lastline";
                chop ($lastline);
            }
        }
    }
  }
  close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 23rd month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "04";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
```

```
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 23rd month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
```

```
    }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

    if ($j == 26) {                  ###final position, check ephem.m file to check line position###
       $R = substr($line,45,8);
       $x = substr($line,80,8);
       $y = substr($line,89,8);
       $z = substr($line,98,8);

       if ($x!=0) {
          $theta = atan($y/$x);
          $theta0 = atan($y0/$x0);
          $delta_theta = $theta - $theta0;

          $delta_y = $R*sin($delta_theta);
          $delta_y = abs($delta_y);
          $delta_z = $z - $z0;

          $R = sprintf("%8.4f",$R);
          $delta_y = sprintf("%8.4f",$delta_y);
          $delta_z = sprintf("%8.4f",$delta_z);

          if ($delta_y>0) {
             $lastline = "$R $delta_y $delta_z 0.00\n";
             printf OF2 "$lastline";
             chop ($lastline);
          }
       }
    }
  }
  close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 24th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "05";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
```

```
        printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
    }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
    open OF, ">epheminputfile1";
    printf OF "i\n";
    printf OF "$styr/$stmonth/$stday1 01:00:00\n";
    printf OF "$endyr/$endmonth/$endday 01:00\n";
    printf OF "$period\n";
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 24th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

        $j++;

        if ($j == 3) {                      ##initial position, check ephem.m file to check line position###
            $R0 = substr($line,45,8);
            $x0 = substr($line,80,8);
                    $y0 = substr($line,89,8);
            $z0 = substr($line,98,8);

            $a0 = 0;

            $R0 = sprintf("%8.4f",$R0);

            $lastline = "$R0 $a0 $a0 0.00\n";
            printf OF1 "$lastline";
            chop ($lastline);
        }

                ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####
```

```
      if ($j == 27) {                  ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 25th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "06";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
```

```
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

####################################
#Extract coords for 25th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####
```

```
      if ($j == 28) {                  ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 26th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "07";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
```

```
        printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
    }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
    open OF, ">epheminputfile1";
    printf OF "i\n";
    printf OF "$styr/$stmonth/$stday1 01:00:00\n";
    printf OF "$endyr/$endmonth/$endday 01:00\n";
    printf OF "$period\n";
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;
printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

####################################
#Extract coords for 26th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

        $j++;

        if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
            $R0 = substr($line,45,8);
            $x0 = substr($line,80,8);
            $y0 = substr($line,89,8);
            $z0 = substr($line,98,8);

            $a0 = 0;

            $R0 = sprintf("%8.4f",$R0);

            $lastline = "$R0 $a0 $a0 0.00\n";
            printf OF1 "$lastline";
            chop ($lastline);
        }

                ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####
```

```
      if ($j == 29) {                    ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 27th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "08";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
```

```
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";


####################################
#Extract coords for 27th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####
```

```
      if ($j == 30) {                  ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);

         if ($x!=0) {
            $theta = atan($y/$x);
            $theta0 = atan($y0/$x0);
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial5.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal5.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial5.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal5.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 28th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "09";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
```

```
    }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
    open OF, ">epheminputfile1";
    printf OF "i\n";
    printf OF "$styr/$stmonth/$stday1 01:00:00\n";
    printf OF "$endyr/$endmonth/$endday 01:00\n";
    printf OF "$period\n";
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 28th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

        $j++;
        if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
            $R0 = substr($line,45,8);
            $x0 = substr($line,80,8);
            $y0 = substr($line,89,8);
            $z0 = substr($line,98,8);

            $a0 = 0;

            $R0 = sprintf("%8.4f",$R0);

            $lastline = "$R0 $a0 $a0 0.00\n";
            printf OF1 "$lastline";
            chop ($lastline);
        }

                ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

        if ($j == 31) {                   ###final position, check ephem.m file to check line position###
```

```
            $R = substr($line,45,8);
            $x = substr($line,80,8);
            $y = substr($line,89,8);
            $z = substr($line,98,8);

            if ($x!=0) {
               $theta = atan($y/$x);
               $theta0 = atan($y0/$x0);
               $delta_theta = $theta - $theta0;

               $delta_y = $R*sin($delta_theta);
               $delta_y = abs($delta_y);
               $delta_z = $z - $z0;

               $R = sprintf("%8.4f",$R);
               $delta_y = sprintf("%8.4f",$delta_y);
               $delta_z = sprintf("%8.4f",$delta_z);

               if ($delta_y>0) {
                  $lastline = "$R $delta_y $delta_z 0.00\n";
                  printf OF2 "$lastline";
                  chop ($lastline);
               }
            }
         }
      }
      close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 29th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "10";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";
```

```
printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 29th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

               $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


      if ($j == 32) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
```

```
            $y = substr($line,89,8);
            $z = substr($line,98,8);

            if ($x!=0) {
               $theta = atan($y/$x);
               $theta0 = atan($y0/$x0);
               $delta_theta = $theta - $theta0;

               $delta_y = $R*sin($delta_theta);
               $delta_y = abs($delta_y);
               $delta_z = $z - $z0;

               $R = sprintf("%8.4f",$R);
               $delta_y = sprintf("%8.4f",$delta_y);
               $delta_z = sprintf("%8.4f",$delta_z);

               if ($delta_y>0) {
                  $lastline = "$R $delta_y $delta_z 0.00\n";
                  printf OF2 "$lastline";
                  chop ($lastline);
               }
            }
         }
      }
      close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 30th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "11";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";
```

```
printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;

   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

###################################
#Extract coords for 30th time step#
###################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

               $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


      if ($j == 33) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
```

```
            $x = substr($line,80,8);
            $y = substr($line,89,8);
            $z = substr($line,98,8);

            if ($x!=0) {
               $theta = atan($y/$x);
               $theta0 = atan($y0/$x0);
               $delta_theta = $theta - $theta0;

               $delta_y = $R*sin($delta_theta);
               $delta_y = abs($delta_y);
               $delta_z = $z - $z0;

               $R = sprintf("%8.4f",$R);
               $delta_y = sprintf("%8.4f",$delta_y);
               $delta_z = sprintf("%8.4f",$delta_z);

               if ($delta_y>0) {
                  $lastline = "$R $delta_y $delta_z 0.00\n";
                  printf OF2 "$lastline";
                  chop ($lastline);
               }
            }
         }
      }
      close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#######################################
#Creation of ephem file for 31st month#
#######################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2005";   #max ending day is 99 days from start day#
$endmonth = "12";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";
```

```
printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 31st month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######

      $j++;
      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 34) {                   ###final position, check ephem.m file to check line positionn###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
```

```
        $z = substr($line,98,8);

        if ($x!=0) {
           $theta = atan($y/$x);
           $theta0 = atan($y0/$x0);
           $delta_theta = $theta - $theta0;

           $delta_y = $R*sin($delta_theta);
           $delta_y = abs($delta_y);
           $delta_z = $z - $z0;

           $R = sprintf("%8.4f",$R);
           $delta_y = sprintf("%8.4f",$delta_y);
           $delta_z = sprintf("%8.4f",$delta_z);

           if ($delta_y>0) {
              $lastline = "$R $delta_y $delta_z 0.00\n";
              printf OF2 "$lastline";

                        chop ($lastline);
           }
        }
     }
  }
  close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 32nd month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2006";   #max ending day is 99 days from start day#
$endmonth = "01";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
  $days = "0$days";
}
else {
  if ($days>99) {
     $days = 31;
     printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
  }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";
```

```
for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$i\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "$month/39 m$i\n";

   `ephem < epheminputfile1`;
   $stmonth2 = 5;
   $stday2 = 1;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 32nd month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;


open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
   open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
   $j = 0;
   while ($line = <IF>) {
 ###Coords are (x0=R0,y0=0,z0=0) ######
      $j++;

      if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
         $R0 = substr($line,45,8);
         $x0 = substr($line,80,8);
         $y0 = substr($line,89,8);
         $z0 = substr($line,98,8);

         $a0 = 0;

         $R0 = sprintf("%8.4f",$R0);

         $lastline = "$R0 $a0 $a0 0.00\n";
         printf OF1 "$lastline";
         chop ($lastline);
      }

            ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

      if ($j == 35) {                   ###final position, check ephem.m file to check line position###
         $R = substr($line,45,8);
         $x = substr($line,80,8);
         $y = substr($line,89,8);
         $z = substr($line,98,8);
```

```
        if ($x!=0) {
           $theta = atan($y/$x);
           $theta0 = atan($y0/$x0);
           $delta_theta = $theta - $theta0;

           $delta_y = $R*sin($delta_theta);
           $delta_y = abs($delta_y);
           $delta_z = $z - $z0;

           $R = sprintf("%8.4f",$R);
           $delta_y = sprintf("%8.4f",$delta_y);
           $delta_z = sprintf("%8.4f",$delta_z);

           if ($delta_y>0) {
              $lastline = "$R $delta_y $delta_z 0.00\n";
              printf OF2 "$lastline";
              chop ($lastline);
           }
        }
     }
  }
  close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 33rd month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2006";   #max ending day is 99 days from start day#
$endmonth = "02";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
```

```
    printf OF "i\n";
    printf OF "$styr/$stmonth/$stday1 01:00:00\n";
    printf OF "$endyr/$endmonth/$endday 01:00\n";
    printf OF "$period\n";
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 33rd month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

        $j++;
        if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
            $R0 = substr($line,45,8);
            $x0 = substr($line,80,8);
            $y0 = substr($line,89,8);
            $z0 = substr($line,98,8);

            $a0 = 0;

            $R0 = sprintf("%8.4f",$R0);

            $lastline = "$R0 $a0 $a0 0.00\n";
            printf OF1 "$lastline";
            chop ($lastline);
        }

                ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####

        if ($j == 36) {                   ###final position, check ephem.m file to check line position###
            $R = substr($line,45,8);
            $x = substr($line,80,8);
            $y = substr($line,89,8);
            $z = substr($line,98,8);

            if ($x!=0) {
                $theta = atan($y/$x);
                $theta0 = atan($y0/$x0);
```

```
            $delta_theta = $theta - $theta0;

            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}
printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;

#####################################
#Creation of ephem file for 34th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2006";   #max ending day is 99 days from start day#
$endmonth = "03";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
```

```
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

##############################
#Extract coords for 34th month#
##############################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

        $j++;

        if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
            $R0 = substr($line,45,8);
            $x0 = substr($line,80,8);
            $y0 = substr($line,89,8);
            $z0 = substr($line,98,8);

            $a0 = 0;

            $R0 = sprintf("%8.4f",$R0);

            $lastline = "$R0 $a0 $a0 0.00\n";
            printf OF1 "$lastline";
            chop ($lastline);
        }

                ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


        if ($j == 37) {                   ###final position, check ephem.m file to check line position###
            $R = substr($line,45,8);
            $x = substr($line,80,8);
            $y = substr($line,89,8);
            $z = substr($line,98,8);

            if ($x!=0) {
                $theta = atan($y/$x);
                $theta0 = atan($y0/$x0);
                $delta_theta = $theta - $theta0;
```

```
            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 35th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2006";   #max ending day is 99 days from start day#
$endmonth = "04";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
```

```
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;

printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

####################################
#Extract coords for 35th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

        $j++;

        if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
            $R0 = substr($line,45,8);
            $x0 = substr($line,80,8);
            $y0 = substr($line,89,8);
            $z0 = substr($line,98,8);

            $a0 = 0;

            $R0 = sprintf("%8.4f",$R0);

            $lastline = "$R0 $a0 $a0 0.00\n";
            printf OF1 "$lastline";
            chop ($lastline);
        }

                ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


        if ($j == 38) {                   ###final position, check ephem.m file to check line position###
            $R = substr($line,45,8);
            $x = substr($line,80,8);
            $y = substr($line,89,8);
            $z = substr($line,98,8);

            if ($x!=0) {
                $theta = atan($y/$x);
                $theta0 = atan($y0/$x0);
                $delta_theta = $theta - $theta0;
```

```
            $delta_y = $R*sin($delta_theta);
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal\$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;


#####################################
#Creation of ephem file for 36th month#
#####################################
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`rm -rf /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions`;
`mkdir /Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path`;
$month++;

$styr = "2003";
$stmonth = "05";  #if changing starting month, must change it down below as well, with no leading 0#
$stday1 = "01";   #if changing starting day, must change it down below as well, with no leading 0#

$endyr = "2006";   #max ending day is 99 days from start day#
$endmonth = "05";
$endday = "01";

$jd1 = julian_day($styr, $stmonth, $stday1);
$jd2 = julian_day($endyr, $endmonth, $endday);
$days = $jd2 - $jd1;
if ($days<10) {
   $days = "0$days";
}
else {
   if ($days>99) {
      $days = 31;
      printf "Days limit interval > 99!  Manually edit data extraction from ephem output file!\n";
   }
}
$period = "$days:00:00";

printf "Starting loop for $obj objects..\n";

for ($i=1; $i<$obj; $i++) {
   open OF, ">epheminputfile1";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$endyr/$endmonth/$endday 01:00\n";
   printf OF "$period\n";
```

```
    printf OF "w\n";
    printf OF "m$i\n";
    printf OF "\n";
    printf OF "\n";
    close OF;
    printf "$month/39 m$i\n";

    `ephem < epheminputfile1`;
    $stmonth2 = 5;
    $stday2 = 1;
    `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/start_endpositions/ephem.m$i`;
}
`rm ./epheminputfile1`;
printf "Loop just ended, results are ephem.XXX in ~/Trident/programming/bias/ephem_results\n";

####################################
#Extract coords for 36th time step#
####################################
$num++;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsinitial$num.out`;
`rm /Users/casparwhittey/Trident/visualization/visgames/bias/ephemorbitsfinal$num.out`;

printf "Extracting R, elcliptic longitude, and ecliptic latitude....\n";
opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF1, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsinitial$num.out";
open OF2, ">/Users/casparwhittey/Trident/programming/bias/ephemorbitsfinal$num.out";

foreach $filename (@resultsfiles) {
    open IF, "/Users/casparwhittey/Trident/programming/bias/ephem_results/start_endpositions/$filename";
    $j = 0;
    while ($line = <IF>) {
  ###Coords are (x0=R0,y0=0,z0=0) ######

        $j++;

        if ($j == 3) {                    ##initial position, check ephem.m file to check line position###
            $R0 = substr($line,45,8);
            $x0 = substr($line,80,8);
            $y0 = substr($line,89,8);
            $z0 = substr($line,98,8);

            $a0 = 0;

            $R0 = sprintf("%8.4f",$R0);

            $lastline = "$R0 $a0 $a0 0.00\n";
            printf OF1 "$lastline";
            chop ($lastline);
        }

                ####Coords are (x=R, delta_y=R*sin(delta_theta), delta_z=z-z0)#####


        if ($j == 39) {                   ###final position, check ephem.m file to check line position###
            $R = substr($line,45,8);
            $x = substr($line,80,8);
            $y = substr($line,89,8);
            $z = substr($line,98,8);

            if ($x!=0) {
                $theta = atan($y/$x);
                $theta0 = atan($y0/$x0);
                $delta_theta = $theta - $theta0;

                $delta_y = $R*sin($delta_theta);
```

```
            $delta_y = abs($delta_y);
            $delta_z = $z - $z0;

            $R = sprintf("%8.4f",$R);
            $delta_y = sprintf("%8.4f",$delta_y);
            $delta_z = sprintf("%8.4f",$delta_z);

            if ($delta_y>0) {
               $lastline = "$R $delta_y $delta_z 0.00\n";
               printf OF2 "$lastline";
               chop ($lastline);
            }
         }
      }
   }
   close IF;
}

printf "final (x,y,z) data stored in ./ephemorbitsinitial$num.out file and in ~/visualization/visgames/bias\n";
printf "final (x,y,z) data stored in ./ephemorbitsfinal$num.out file and in ~/visualization/visgames/bias\n";

close OF1;
close OF2;

`mv ephemorbitsinitial$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;
`cp ephemorbitsfinal$num.out /Users/casparwhittey/Trident/visualization/visgames/bias`;



#############################################################
#Creation of another dx data set that shows path of objects#
#############################################################

#set the length of days to run orbit path..

$length = 365*5;
$jd = $length + $jd1;
($yr, $mon, $day) = inverse_julian_day($jd);
$yr = sprintf("%04d",$yr);
$mon = sprintf("%02d",$mon);
$day = sprintf("%02d",$day);


$period = "05:00:00";

printf "Starting ephem loop for orbital path datafile....\n";

for ($i=0; $i<100; $i++) {

   $randobject = int rand($obj);

   open OF, ">epheminputfile2";
   printf OF "i\n";
   printf OF "$styr/$stmonth/$stday1 01:00:00\n";
   printf OF "$yr/$mon/$day 01:00\n";
   printf OF "$period\n";
   printf OF "w\n";
   printf OF "m$randobject\n";
   printf OF "\n";
   printf OF "\n";
   close OF;
   printf "computing orbits for:\n";
   printf "m$randobject\n";

   `ephem < epheminputfile2`;
   `mv /sw/tmp/ephem.$stmonth2-$stday2-$styr ./ephem_results/orbit_path/ephem.m$randobject`;
}
`rm ./epheminputfile2`;
```

```
#################################################################
#extracting info from ephem files to create orbit path dataset#
#################################################################
printf "Extracting info for creatioin of orbit path datafiles...\n";

opendir THISDIR, "/Users/casparwhittey/Trident/programming/bias/ephem_results/orbit_path/";
@resultsfiles = grep /ephem.m/, readdir THISDIR;
closedir THISDIR;

open OF, ">./THISFILE";

foreach $filename (@resultsfiles) {
   `awk 'NF==17{print$5}' './ephem_results/orbit_path/$filename' >> test`;
   open IF, "/Users/casparwhittey/Trident/programming/bias/test";
   while ($line = <IF>) {
            if ($line =~ "UT") {
            }
            else {
               printf OF "$line";
            }
            if ($line =~ "Brightest") {
            }
            else {
               printf OF "$line";
            }
   }

   `rm ./test`;
   close IF;
}
close OF;

`rm /Users/casparwhittey/Trident/visualization/visgames/bias/orbitpathdata.out`;

open OF, ">/Users/casparwhittey/Trident/visualization/visgames/bias/orbitpathdata.out";
open IF, "/Users/casparwhittey/Trident/programming/bias/THISFILE";

while ($line = <IF>) {
   $x = substr($line,80,8);
   $y = substr($line,89,8);
   $z = substr($line,98,8);

   $x = sprintf("%8.4f",$x);
   $y = sprintf("%8.4f",$y);
   $z = sprintf("%8.4f",$z);

   $lastline = "$x $y $z 0.00\n";
   printf OF "$lastline";
   chop ($lastline);
}
close IF;
close OF;

`rm ./THISFILE`;
printf "orbit path (x,y,z) data stored in ~/Trident/visualizations/visgames/bias/orbit_pathdata\n";


#################################################################################
#0        1        2        3        4        5        6        7        8        9        1        1        2
#012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345
67890123456789
#
# 2005 Nov  1  1:00c 19 00 17.55 -17 47 23.1  51.0162  51.3265 289.1319   4.3324  17.5568 -47.7429   3.8773
# 2005 Nov  6  1:00n 19 00 30.07 -17 47 21.9  51.0176  51.4078 289.1813   4.3259  17.5680 -47.7402   3.8776
# 2005 Nov 11  1:00n 19 00 44.46 -17 47 16.0  51.0190  51.4861 289.2383   4.3197  17.5793 -47.7375   3.8780
# 2005 Nov 16  1:00n 19 00  0.62 -17 47  5.2  51.0204  51.5608 289.3024   4.3138  17.5905 -47.7348   3.8783
# 2005 Nov 21  1:00n 19 00 18.46 -17 46 49.7  51.0218  51.6313 289.3733   4.3083  17.6018 -47.7321   3.8786
```

```
#
# Year Mon DY  Time  RH RM RSSSS DDD DM DSSS   RSun     Delta  LONGITUD LATITUDE   X(AU)    Y(AU)    Z(AU)

exit ;
```

Appendix C: Line of Sight Program Code

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

/* Definitions */

#define T0 53451.5
#define PI 3.14159265358279
#define RAD2DEG 180.0/PI
#define DEG2RAD PI/180.0
#define EARTHRATE 360.0/365.25*DEG2RAD
#define MINDIST 0.0
#define MAXDIST 98.0
#define X0 100.0
#define Y0 100.0
#define Z0 100.0
#define ALPHA 0.01617 /* Angular radius of observing area as circle */
#define INCREMENT 0.5
#define PIXSCALE 1.0
#define DEFAULT_EFFICIENCY 0.90

/* Prototyping subroutine */

int earthpos(double t,double *xe,double *ye,double *ze);
int newpix_unique(int x, int y, int z, int *tagx, int *tagy, int *tagz,
        int *ntag);

main() {

        int i,j,k,xmin,xmax,ymin,ymax,zmin,zmax,ix,iy,iz;
    int tagx[1000],tagy[1000],tagz[1000],ntag,mycount;
    int nmonths, lastmonth;
    float fraction[1000];
        float data=5;
        float *ptr_to_data; /* This is not a variable, it is a reference to a block of memory */
        float **ptr_to_ptr; /* This says that we have a pointer to a pointer to a float */
        float ***bigarray;  /* This is a big 3d array */
    double t,l,b,xe,ye,ze,ax,ay,az,r,xlos,ylos,zlos,rcone,rtest;
    double deltax,deltay,deltaz,starttime=0;
        FILE *INFILE, *OUTFILE1, *OUTFILE2;

        /* here is the 3-d you need to make, try 201 */

        bigarray = (float***) malloc(201 * sizeof(float**));
        if(bigarray == NULL) {
          printf("Malloc for 3-d didn't work!\n");
          exit(1);
        }
        for (i = 0; i<201; i++) {
          bigarray[i] = (float**) malloc(201 * sizeof(float*));
          if (bigarray[i] == NULL) {
            printf("Malloc for 3-d step 2 didn't work!\n");
            exit(1);
          }
          for (j=0; j<201; j++) {
            bigarray[i][j] = (float*) malloc(201 * sizeof(float));
            if (bigarray[i][j] == NULL) {
              printf("Malloc for 3-d step 3 didn't work!\n");
              exit(1);
            }
          }
        }

        /* Arrays should always be initialized, this array is dimensions 101 X 101 X 101 */
```

```
                for (i=0; i<201; i++) {
                  for (j=0; j<201; j++) {
                    for (k=0; k<201; k++) {
                      bigarray[i][j][k] = 0;
                    }
                  }
                }

/* Read the pointing history file and process it one line at a time */

      if ((INFILE=fopen("./pointing_history.dat","r")) == NULL) {
        printf("No pointing history file in directory.\n");
                exit(1);
      }
      if ((OUTFILE1=fopen("./earthpos.dat","w")) == NULL) {
        printf("Cant output in this directory.\n");
                exit(1);
      }
      if ((OUTFILE2=fopen("./los.dat","w")) == NULL) {
        printf("Cant output in this directory.\n");
                exit(1);
      }

      while(!feof(INFILE)) { /* Same as while($line = <IF>) */

        fscanf(INFILE,"%lf%lf%lf",&t,&l,&b);

        if (starttime==0) starttime = t;

        nmonths = (int) ((t-starttime)/30);
        if (nmonths != lastmonth) {
          printf("In month %d\n",nmonths);
          lastmonth = nmonths;
        }

        /* Find earth */
        earthpos(t,&xe,&ye,&ze);
        fprintf(OUTFILE1,"%lf %lf %lf %lf\n",t,xe,ye,ze);

        /* Need to find the slopes of the line of sight */

        ax = cos(DEG2RAD*l)*cos(DEG2RAD*b);
        ay = sin(DEG2RAD*l)*cos(DEG2RAD*b);
        az = sin(DEG2RAD*b);

                /* Need to step out on line of sight in INCREMENT pixel increments */

        ntag=0;

        for (r = MINDIST; r < MAXDIST; r+=INCREMENT) {

          xlos = ax*r + xe;
          ylos = ay*r + ye;
          zlos = az*r + ze;

/*  fprintf(OUTFILE2,"%lf %lf %lf %lf\n",t,xlos,ylos,zlos); */

          /* Find the physical radius of the observing cone */

          rcone = ALPHA * r;

                  /* Now we have to determine ranges of coordinates for pixels */

          xmin = (int)(xlos - rcone - INCREMENT -1);
          xmax = (int)(xlos + rcone + INCREMENT +1);
          ymin = (int)(ylos - rcone - INCREMENT -1);
          ymax = (int)(ylos + rcone + INCREMENT +1);
          zmin = (int)(zlos - rcone - INCREMENT -1);
```

```
        zmax = (int)(zlos + rcone + INCREMENT +1);

        for (ix = xmin; ix < xmax; ix++) {
          deltax = (ix - xlos)*(ix - xlos);
          for (iy = ymin; iy < ymax; iy++) {
            deltay = (iy - ylos)*(iy - ylos);
            for (iz = zmin; iz < zmax; iz++) {
              deltaz = (iz - zlos)*(iz - zlos);
              rtest = sqrt(deltax+deltay+deltaz);
                              if (rtest < rcone + INCREMENT) {
                if (newpix_unique(ix,iy,iz,tagx,tagy,tagz,&ntag)) {
                  fprintf(OUTFILE2,"%lf %d %d %d\n",t,ix,iy,iz);
                }
                if (fabs(rtest-rcone)<0.5) {
                  if (rtest > rcone) {
                    fraction[ntag-1] = fabs(rtest-rcone);
                  }
                  else {
                    fraction[ntag-1] = 1.0-fabs(rtest-rcone);
                  }
                }
                else {
                  fraction[ntag-1] = 1.0;
                }
              }
            }
          }
        }


      }

      /* Insert the unique pixels into bigarray, being mindful of partials */

      for (ix = 0; ix < ntag; ix++) {
                bigarray[tagx[ix]][tagy[ix]][tagz[ix]] =
                  bigarray[tagx[ix]][tagy[ix]][tagz[ix]] +
                  (1 - bigarray[tagx[ix]][tagy[ix]][tagz[ix]])*fraction[ix]*DEFAULT_EFFICIENCY;
      }


    }
    fclose(INFILE);
    fclose(OUTFILE1);
    fclose(OUTFILE2);

    mycount = 0;

    return 0;

}


int earthpos(double t,double *xe,double *ye,double *ze) {

  *ze = Z0;
  *xe = (1/PIXSCALE)*cos(EARTHRATE*(t-T0))+X0;
  *ye = (1/PIXSCALE)*sin(EARTHRATE*(t-T0))+Y0;

  return 0;

}

int newpix_unique(int x, int y, int z, int *tagx, int *tagy, int *tagz,
      int *ntag) {

   int i, unique;
```

```
  unique = 1;


  for (i=0; i<(*ntag); i++) {
//    printf("Testing %d (%d,%d,%d) to (%d,%d,%d) -- listsize %d\n",i,x,y,z,tagx[i],tagy[i],tagz[i],*ntag);
    if (x==(tagx[i]) && y == (tagy[i]) && z == (tagz[i])) unique = 0;
  }

  if (unique) {
    tagx[(*ntag)]=x;
    tagy[(*ntag)]=y;
    tagz[(*ntag)]=z;
    (*ntag)++;
  }

  return unique;

}
```