

Master's Thesis
Astronomy

Asteroid identification at discovery

Mikael Granvik

2003

Tutor and censor: Doc. Karri Muinonen

Censor: Doc. Hannu Karttunen

UNIVERSITY OF HELSINKI
DEPARTMENT OF ASTRONOMY
P.O. Box 14 (Kopernikuksentie 1 A)
FIN-00014 University of Helsinki

Tiedekunta/Osasto — Fakultet/Sektion — Faculty		Laitos — Institution — Department	
Matematisks-naturvetenskapliga		Astronomiska institutionen	
Tekijä — Författare — Author			
Mikael Granvik			
Työn nimi — Arbetets titel — Title			
Asteroid identification at discovery			
Oppiaine — Läroämne — Subject			
astronomi			
Työn laji — Arbetets art — Level		Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
pro gradu		december 2003	65
Tiivistelmä — Referat — Abstract			
<p>I arbetet presenteras en ny metod för identifikationen av asteroider vid tidpunkten för upptäckten och under de närmaste därpå följande dagarna. Identifikationsproblemet för asteroider uppstår när man ur två skilda grupper av observationer försöker hitta de gemensamma asteroiderna som observerats vid bägge tillfällena. I början vet man inte ens om det överhuvudtaget finns några gemensamma asteroider bland observationerna. Den nya algoritmen är den enda kända metoden som löser identifikationsproblemet för nyligen upptäckta asteroider.</p> <p>I praktiken går man tillväga genom att gallra bort omöjliga identifikationer i en flerstegsprocess. Först jämförs efemerider och deras differenser inom en bestämd tidsintervall. Identifikationsalgoritmen använder sig av en inversionsteknik kallad statistisk variering med vilken det är möjligt att härleda banelementens fördelning utgående från endast två observationer. Den stora mängden data, som i tidigare tillämpningar varit ett problem hos denna Monte Carlo -metod, har i detta arbete effektivt undvikits genom användningen av en sk. adressjämförelseteknik. Tekniken går ut på att man transformerar realtalen (här efemeriderna och deras differenser) till ett enda heltal och använder detta heltal för att hitta liknande lösningar. Till slut försöker man framställa banfördelningar åt möjliga identifikationer. Ifall det lyckas, bevisar det att det är möjligt att koppla ihop observationerna inom ramarna för deras felmarginaler.</p> <p>Identifikationsproblemet har undersökts och den nya metoden har testats med simulerade observationer av när- och huvudbältsasteroider. När algoritmen testades med observationer producerade enligt olika observationsstrategier, upptäcktes det att Minor Planet Centers rekommendationer inte nödvändigtvis ger optimala resultat med tanke på identifikationsproblemet.</p> <p>I fortsättningen är det meningen att bl.a. optimera parametrarna för att möjliggöra effektiv identifikation, undersöka hur olika observationsgeometrier inverkar på identifikationen, försöka identifiera asteroider bland hittills oidentifierade riktiga observationer och leta efter en optimal observationsstrategi med tanke på asteroiders identifikation.</p>			
Avainsanat — Nyckelord — Keywords			
celest mekanik, asteroid, identifikation, simulerade observationer, NEO, MBO			
Säilytyspaikka — Förvaringsställe — Where deposited			
Astronomiska institutionens bibliotek			
Muita tietoja — övriga uppgifter — Additional information			

Tiedekunta/Osasto — Fakultet/Sektion — Faculty		Laitos — Institution — Department	
Matemaattis-luonnontieteellinen		Tähtitieteen laitos	
Tekijä — Författare — Author			
Mikael Granvik			
Työn nimi — Arbetets titel — Title			
Asteroid identification at discovery			
Oppiaine — Läroämne — Subject			
tähtitiede			
Työn laji — Arbetets art — Level		Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
pro gradu		joulukuu 2003	65
Tiivistelmä — Referat — Abstract			
<p>Työssä esitellään uusi ja toistaiseksi ainoa tunnettu menetelmä asteroidien lyhyen aikavälin tunnistusongelman ratkaisemiseksi välittömästi uuden löydön tapahtuessa ja muutamien löytöä seuraavien päivien aikana. Asteroidien tunnistusongelma syntyy, kun kahdesta erillisestä havaintojoukosta yritetään poimia esiin molemmissa esiintyvät kohteet. Alussa ei ole tiedossa, onko ylipäätään olemassa kohteita, jotka esiintyvät kummassakin joukossa.</p> <p>Käytännössä ongelmaa lähdetään ratkaisemaan monivaiheisesti vertailemalla ensin eri havaintojoukkojen kohteille laskettuja taivaankannen koordinaatteja sekä näiden muutosta tietyn aikavälin kuluessa. Tunnistusalgoritmin perustana käytetään tilastolliseksi luotaukseksi kutsuttua inversiomenetelmää, jolla voidaan johtaa rataelementtien jakauma jo kahden havainnon perusteella. Aiemmissä sovelluksissa tämän Monte Carlo -tyyppisen inversiomenetelmän ongelmana ollut suuri datamäärä kyetään tässä työssä käsittelemään tehokkaasti nk. osoitevertailumenetelmällä, jossa reaalityypit (tässä rektaskensio, deklinaatio sekä näiden muutos) muunnetaan yhdeksi kokonaisluvuksi ja samantyyppiset ratkaisut löydetään vertailemalla näitä kokonaislukuja. Lopuksi mahdolliselle kytkennälle yritetään laskea ratajakauma, joka onnistuessaan osoittaa tunnistuksen mahdolliseksi havaintovirheiden sallimissa rajoissa.</p> <p>Tunnistusongelmaa on tutkittu ja uutta menetelmää testattu simuloitulla lähi- ja päävyöhykkeen asteroidien havainnoilla. Testattaessa algoritmia eri havaintostrategioiden suositusten perusteella simuloituilla havainnoilla huomattiin, että Minor Planet Centerin suositukset eivät välttämättä ole optimaaliset tunnistusongelmaa ajatellen.</p> <p>Jatkossa on tarkoitus mm. etsiä sopivimmat parametrit tehokasta tunnistusta varten, tutkia erilaisten havaintogeometrioiden vaikutusta tunnistukseen, ajaa tunnistusalgoritmi toistaiseksi tunnistamattomille oikeille havainnoille, sekä etsiä optimaalista havaintostrategiaa tunnistusongelman näkökulmasta.</p>			
Avainsanat — Nyckelord — Keywords			
taivaanmekaniikka, asteroidi, tunnistus, simuloitut havainnot, NEO, MBO			
Säilytyspaikka — Förvaringsställe — Where deposited			
Observatorion kirjasto			
Muita tietoja — övriga uppgifter — Additional information			

Abstract

A novel method, the first of its kind published, for the identification of asteroids at discovery and few days thereafter, is presented. Having two different sets of observations, the identification problem arises when one tries to determine which objects reside in both sets. It is not even known *a priori*, whether the sets have objects in common at all.

The identification problem is solved in a multistep filtering process, which first compares ephemeris positions and motions. The algorithm is built upon an orbit inversion technique, termed statistical ranging, which allows rigorous mapping of the orbital element distribution using two or more observations. The huge output of data, which is a common problem of Monte Carlo methods, is here avoided by using the so-called address comparison technique. The idea is to transform an array of real numbers (here, the position and motion) to a single integer, which is used in the search for similar values.

Finally, an orbit distribution is computed for a possible linking. If the inversion succeeds, it shows that an orbit exists, thus tying the observations together assuming predefined observational errors.

The new method presented here is tested with simulated observations of near-Earth and main-belt objects. The tests, which are the most accurate and extensive published to date, show that the method is functional, as expected. Running the algorithm on observational sets simulated according to different observation strategies shows, that the strategy promoted by the Minor Planet Center might not be the best one from an identification point of view.

In the future, several areas must be addressed, such as establishing optimised parameters for the identification procedure, examining the effect that different observing geometries have on identification, scanning through archives of real unidentified observations, and looking for an observation strategy that is optimised for identification.

Contents

1	Introduction	1
1.1	Asteroids observed	1
1.2	Solving the identification problem	6
1.3	Terminology	7
1.4	Aim of the thesis	7
2	Modern orbit determination	9
2.1	Two-body problem	9
2.2	f - and g -functions	10
2.3	p -iteration method for solving an orbit	12
2.4	Criteria for acceptance	13
3	Algorithm for identification at discovery	15
3.1	Statistical orbital ranging	16
3.2	Removing the bulk of impossible candidates	17
3.2.1	Asymmetric ephemeris comparison	17
3.2.2	Phase space address comparison	18
3.2.3	Symmetric ephemeris address comparison	21
3.3	Sieving possible linkings	22
3.4	Finding potential identifications	22
4	Simulated observations	25
4.1	ASurv simulation software	26
4.2	Generating simulated observations	26
4.3	Observational properties of the simulated asteroid population	30
4.3.1	Verification	30
4.3.2	Statistical opposition effect	30
4.3.3	Positions and motions	30
5	Results and discussion	40
5.1	Adjusting the parameters	40
5.1.1	First step — Symmetric ephemeris address comparison	40

5.1.2	Second step — Single-orbit filtering	43
5.1.3	Third step — Orbit distribution filtering	46
5.2	Identification of single-night asteroids	47
5.3	Cumulative identification and observational strategies	49
5.3.1	Minor Planet Center	50
5.3.2	Automatic surveys	51
5.4	Interpretation	52
6	Conclusions	54
	References	56
A	Array-integer transformation algorithms	59
A.1	Transforming an array to an integer	59
A.2	Transforming an integer to an array	61
B	Algorithms for sorting and searching	63
B.1	Insertion sort	63
B.2	Binary search	64

Chapter 1

Introduction

A novel and the first published method for identifying asteroids at discovery and for a few days thereafter, is presented. Having two different sets of observations, the identification problem arises when one tries to find out which objects reside in both sets (Figure 1.1). It is not known *a priori*, whether the sets have objects in common at all.

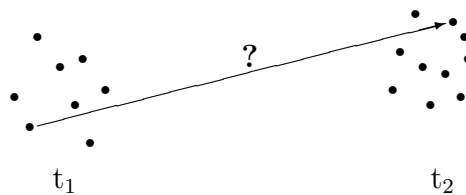


Figure 1.1: The goal of asteroid identification is to link observations of an object observed at one epoch (t_1) to the observations of the same object at another epoch (t_2). Here, the observational sets contain observations of several objects, which makes the problem more realistic, and also more difficult. It is not usually known *a priori* whether the sets have objects in common at all.

1.1 Asteroids observed

The definition of an asteroid is somewhat vague. This is due to the problem that arises when one compares asteroids with planets or dust grains, because strict physical rules to distinguish between these three do not exist. Contrary to this, the path from microscopic grains up to gas giants is, more or less, a smooth continuum. An asteroid can, however, be defined as a macroscopic body on a heliocentric orbit whose own gravitational force is usually too weak to make the body spherical. The diameters of asteroids range from a few metres up to hundreds of kilometres. As a rule of thumb, asteroids get larger with increasing

distance from the Sun. The largest near-Earth objects (NEOs) have diameters of a few tens of kilometres, while some transneptunian objects (TNOs), residing at the outer edges of the solar system, have been measured to be one thousand kilometres across.

The distribution of asteroids in the solar system, however, is not continuous. If traveling outwards from the Sun, the first population to be encountered are the NEOs. They are defined as asteroids having perihelion distances $q < 1.3$ AU, and aphelion distances $Q > 0.983$ AU. Some NEOs, forming a subcategory called Atens, even have semimajor axes, a , smaller than that of Earth's ($a_{\oplus} = 1$ AU). Due to relatively high eccentricities, the orbits of Atens are not completely inside Earth's orbit, but there is no, e.g., dynamical reason why they could not be. There are only a few objects found to date that might orbit the Sun constantly within Earth's orbit. The probable reason for such a low number of inner-Earth objects (IEOs) is the demanding observing geometry. Besides Atens, the three subcategories of the NEO population also include the Apollos ($a \geq 1.0$ AU and $q \leq 1.0167$ AU) and the Amors (1.0167 AU $< q \leq 1.3$ AU).

The main asteroid belt between Mars and Jupiter contains the bulk of discovered minor planets. These asteroids are termed main-belt objects, or MBOs. Because of their observational properties, they were the first asteroids to be discovered¹ and has until recent years been the best observed asteroid population. MBOs are the easiest asteroids to detect because their motions are neither too slow to be noticed, nor too fast to slip away. On the other hand, the relatively large diameters of MBOs, combined with reasonable distances, keep their apparent magnitudes on an easily detectable level.

Another group closely related to MBOs are the Jupiter Trojans. Their orbits are similar to Jupiter's, and they are either leading or trailing the gas giant at angular distances of approximately 60° . The two areas, where these objects can be found, are centered at the Lagrangian L_4 and L_5 points of the Sun-Jupiter-system. All known NEOs, well-known MBOs and Jupiter Trojans, and all known comets within the area, are plotted in Figure 1.2 according to the situation on November 11th, 2003.

There are two main groups of asteroids outside Jupiter's orbit, namely Centaurs and TNOs. The TNO population was first discovered by Jewitt & Luu (1993). So-called pencil-beam surveys, which are able to detect faint targets with a trade-off in coverage, are used to search for these bodies. TNOs are practically residing on the same spots where they were found, due to a very slow apparent motion and a recent discovery during the last decade. The footprints

¹Asteroid Ceres, the largest MBO with a diameter of some 946 km, was discovered in 1801 by Giuseppe Piazzi.

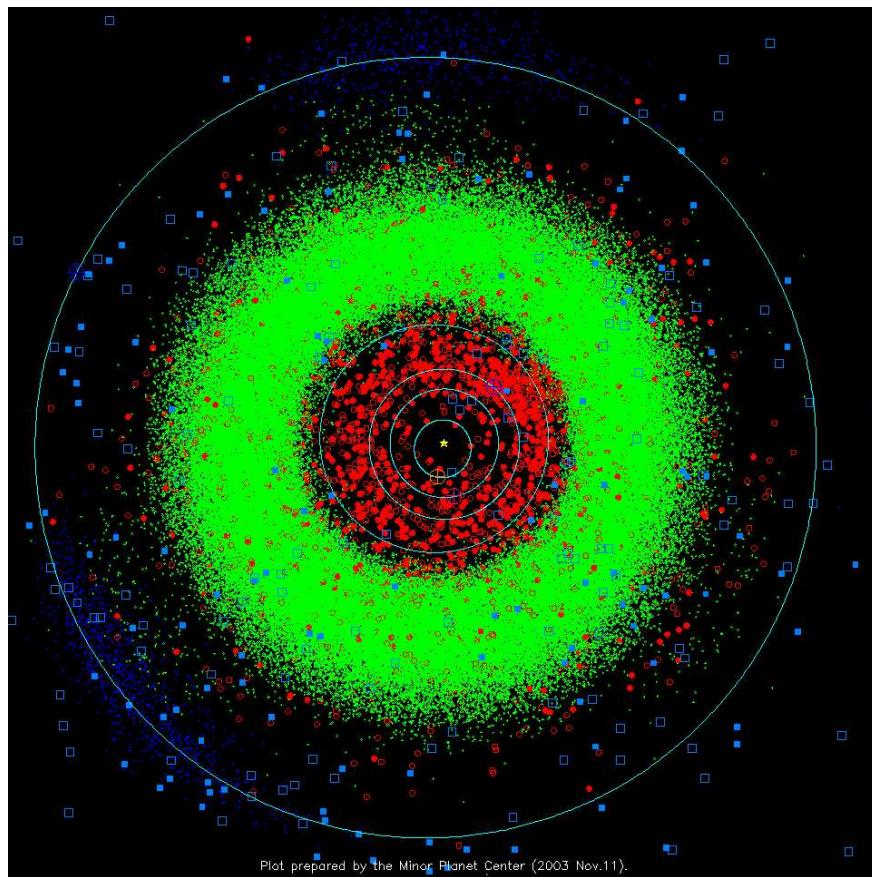


Figure 1.2: The inner solar system on November 11th, 2003. Orbits of the five innermost planets (Mercury, Venus, Earth, Mars, and Jupiter) are shown in light blue, with large coloured dots indicating the current positions. NEOs are indicated by red circles, while numbered and unnumbered, but reasonably well-known MBOs are marked with green circles. The two “clouds” of objects, here coloured deep blue, 60° ahead and behind Jupiter, are Jupiter Trojans. The filled, light-blue squares indicate numbered periodic comets, while other comets are shown as unfilled, light-blue squares. The plot has been prepared by the Minor Planet Center.

of the pencil-beam surveys are thus still apparent in Figure 1.3, which shows the outer solar system as of November 11th, 2003.

So far 232,470 minor planets² have been discovered and 73,636 of these have been numbered, i.e., they have been observed enough to allow for computation of reasonably accurate orbital elements. Reasonably accurate orbital elements refers to the situation when an asteroid’s position can be determined with a precision more accurate than $1''$ during a period of ten years without any further observations.

An asteroid is observed during a so-called apparition³, which occurs around

²This was the situation on November 9, 2003 according to the Minor Planet Center at <http://cfa-www.harvard.edu/iau/mpc.html>.

³The relatively short period when an object is continuously observable.

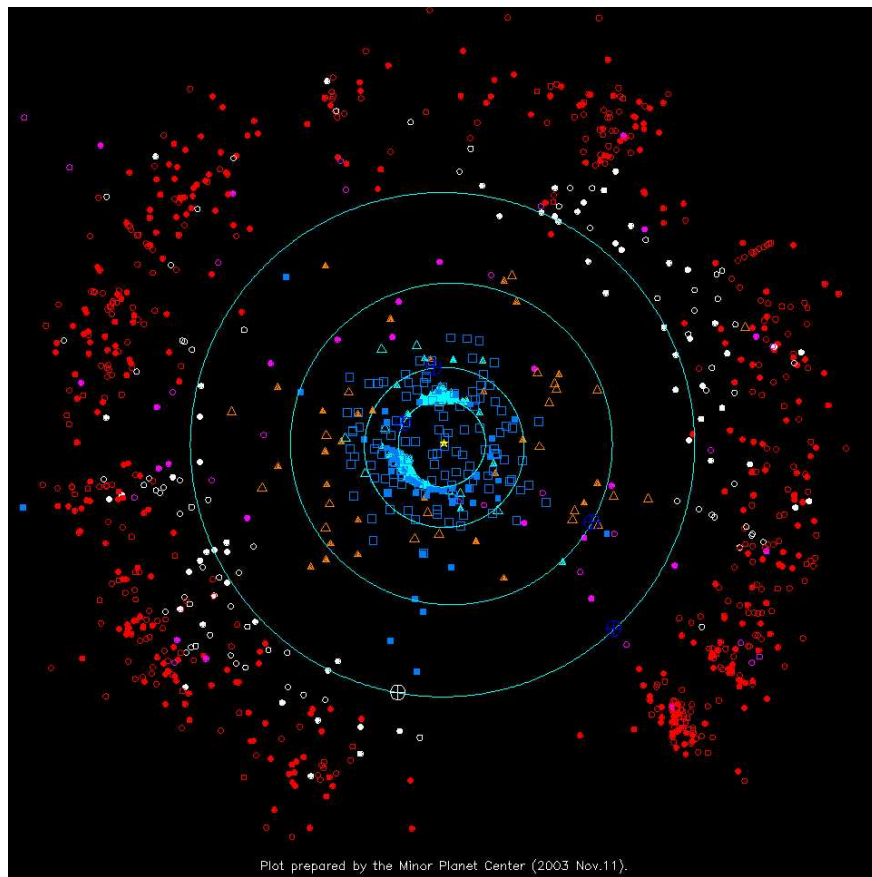


Figure 1.3: The outer solar system on November 11th, 2003. Orbits of the giant planets (Jupiter, Saturn, Uranus, and Neptune) are shown in light blue, with large deep blue dots indicating their current positions. Pluto is just marked with a large white dot. The two “clouds” of objects, here coloured light blue, 60° ahead and behind Jupiter, are Jupiter Trojans. Centaur objects are marked with orange triangles. Different subcategories of TNOs are indicated by red, white, and magenta coloured circles. Unusual high-eccentricity objects are marked with cyan triangles. Filled, light-blue squares indicate numbered periodic comets, while other comets are shown as unfilled, light-blue squares. The plot has been prepared by the Minor Planet Center.

the time of the opposition of the object. The time interval between two oppositions is typically one year; the closer an asteroid is to Earth, the more the interval differs from the typical value. These circumstances lead to the fact that asteroids cannot be observed continuously during a complete revolution. The observational data is thus split into several short pieces.

Given two different sets of observations, each of which spans over an arbitrary observational arc, it is unknown *a priori* whether they refer to the same object or not. This is the basic formulation of the identification problem. Since orbit computation for bodies with short observational arcs is difficult, linking observations between two apparitions is a nontrivial task and the *long-term* identification

problem becomes evident.

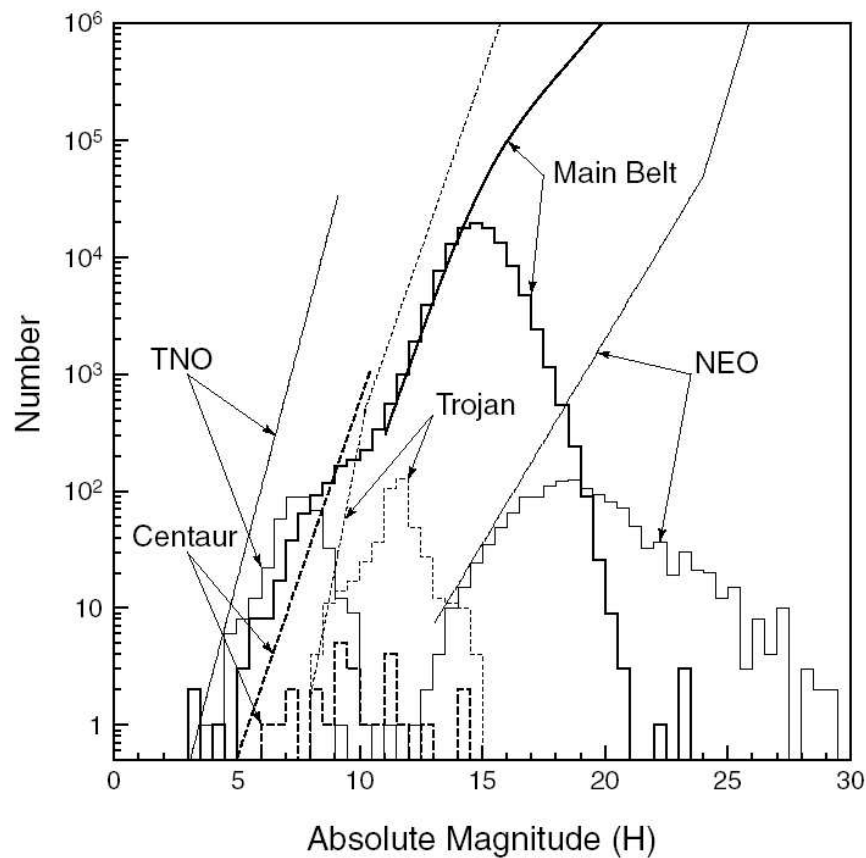


Figure 1.4: Differential H -distribution for five different asteroid populations (Jedicke et al. 2003). Bias-corrected estimates for each population are displayed as lines of various types. Each line merges toward the left of the figure with the known distribution of asteroids as of July 18, 2001. $H = 3^m$ equals a diameter of approximately 670–1500 kilometres, while $H = 30^m$ corresponds to 3–6 metres.

Jedicke et al. (2003) estimated the total number of NEOs and MBOs with diameters larger than 1 km (corresponding to an absolute magnitude $H \lesssim 18^m$) to be about 1,000 and 350,000, respectively. The number of asteroids grows exponentially with decreasing size and increasing H , as seen in Figure 1.4. As the limiting magnitude of telescopes approaches a value of 24^m , e.g., the LSST⁴ and the Pan-STARRS⁵, the number of observed asteroids per unit area will virtually explode. For these forthcoming deep surveys of asteroids, the *short-term* identification of observations within a single apparition becomes a highly challenging problem. Confusion typically arises whether given sets of observations separated by just a few days can be safely addressed to a single asteroid or not.

⁴Large Synoptic Survey Telescope, see <http://www.lsst.org/>

⁵Panoramic Survey Telescope And Rapid Response System, see <http://pan-starrs.ifa.hawaii.edu/>

1.2 Solving the identification problem

Bowell et al. (2003) divided the asteroid orbit computation problem into the inverse and prediction problems. The inversion from observations, usually Right Ascensions (R.A.) and Declinations (Dec.), to orbital elements, was for the first time solved in a practical way by Gauss more than two centuries ago (1801). Since then the goal of the inversion has been to find a set of orbital elements fitting best to observed positions. In the book *Asteroids II*, published in 1989, the only authors referring to the ephemeris uncertainty were Bowell et al. (1989) and Ostro (1989). A few years later, orbital element uncertainty estimations were introduced as a part of the state of the art inversion methods. Muinonen & Bowell (1993) laid out the theoretical basis for a Bayesian treatment of the orbit computation problem, and used the first nonlinear approach on objects with long observational arcs. Recently, Virtanen et al. (2001) and Muinonen et al. (2001) presented an orbit computation method, termed statistical ranging, based on the theory laid out by Muinonen & Bowell. Statistical ranging is particularly suitable for objects with short observational and orbital arcs, e.g., recently discovered NEOs and the majority of the known TNOs (Virtanen et al. 2003). Compared to other orbit inversion techniques, statistical ranging is the only method capable of dealing with strongly nonlinear cases (Milani 1999, Milani & Valsecchi 1999, Chesley 2003).

When the inversion has been carried out successfully, the orbital parameters can be used for solving different prediction problems, such as ephemeris prediction and identification. Milani (1999) divided the identification problem into different categories based on the success of the inversion from observations to orbital elements:

- The data is sufficient to allow for an orbit solution for both sets separately.
- An orbit solution is found separately for one set only. In this case the orbit solution of the first set is compared to the data of the second set, which is insufficient to allow for an orbit solution on its own.
- Both arcs are insufficient to allow for separate orbit solutions. In this case the data of both arcs joined together is, however, sufficient to allow for an orbit solution.

The suitability of the above categorisation is discussed further in the final chapter.

Present-day identification algorithms rely on linear or semilinear approximations (e.g. Milani et al. 2000, Milani et al. 2001). Exact nonlinear algorithms however, like statistical ranging, offer a rigorous foundation to build upon. Since statistical ranging can be applied to objects having only two observations, the

Term	Explanation
angular observational arc	Apparent angular arc from the first to the last observation.
false positive [identification] identification	An identification that is possible, but false. Connecting two observational sets to a single object. Also: linking, linkage
observational scheme	A series of numbers indicating the observation night as days from discovery.
observational set	A collection of observations of one or more objects.
observational [time] arc	Time span from the first to the last observation.
orbital arc	True observed arc; $\Delta\nu$, where ν is the true anomaly.
possible identification	An identification that has not been proven false.

Table 1.1: Terminology related to the identification problem.

identification algorithm built upon it can also be used when two positions become distinguishable, i.e., as soon as it is clear that an object is moving. This has not been possible with previous methods due to the inversion techniques used and the aforementioned approximations in the identification algorithms.

1.3 Terminology

There have been too few articles dedicated to the asteroid identification problem in referred journals. Consequently, the terminology has not yet converged, making it difficult to classify various aspects of the problem. The terminology used in this thesis, with the corresponding explanation, is presented in Table 1.1.

1.4 Aim of the thesis

The ultimate goal of the ongoing research is to produce a set of identification algorithms jointly capable in dealing with arbitrary observational arcs as well as any number of observations. The targets range from NEOs to TNOs, and the set will be particularly suitable for large scale surveys such as European Space Agency's GAIA-mission and the Lowell Observatory Near-Earth-Object Search (LONEOS). The LONEOS project will become an even more interesting

collaborator as one of the main users of the coming Discovery Channel Telescope⁶.

The aim of the thesis is to study the identification problem in connection with objects displaying very short observational and orbital arcs. A decision was made to focus on the NEO and MBO populations, because several automatic NEO surveys, that are currently running, could benefit from the results. Whether NEOs and MBOs can be erroneously linked together during an identification process, provides an example of the many unresolved questions in this area of research.

This thesis presents the first systematic and realistic study on how parameters, such as observational time arc between observational sets, and limiting magnitude, affect the identification problem. In other words, the aim is not to look only for a solution to the short-term identification problem, but also to study different aspects of the problem.

The thesis is organised as follows. Chapter 2 covers modern orbit determination theory, while in Chapter 3 the statistical ranging technique is described, and the building blocks of an identification algorithm are presented. In Chapter 4 the asteroid survey simulation software for generating observations is outlined. Some observational properties of the generated population are also included. In Chapter 5, results produced by the identification algorithm are presented and discussed. Finally, in Chapter 6, the key findings are summarised.

⁶See <http://www.lowell.edu/dct/dct.php>

Chapter 2

Modern orbit determination

2.1 Two-body problem

In addition to the three laws of motion, Newton, in his *Principia*, introduced the law of universal gravitation:

$$\mathbf{F} = G \frac{m_1 m_2}{r^2} \frac{\mathbf{r}}{|\mathbf{r}|}. \quad (2.1)$$

Newton's law of gravity expresses how particles act when impressed by forces, and this still remains the fundamental law of modern celestial mechanics.

According to Newton's three laws of motion and the law of universal gravitation (2.1), a system with $N + 1$ bodies can be described with the following equations of motion:

$$\frac{d^2 \mathbf{r}_i}{dt^2} = \gamma \sum_{j=0, j \neq i}^N m_j \frac{\mathbf{r}_{ij}}{r_{ij}^3}, \quad (i = 0, N), \quad (2.2)$$

where γ is the universal constant of gravity, \mathbf{r}_i are the Cartesian position vectors of the $N + 1$ bodies and $\mathbf{r}_{ij} = \mathbf{r}_j - \mathbf{r}_i$ their relative position vectors. Three assumptions have to be fulfilled when using the equation. Firstly, the bodies have to be spherically symmetric with their masses concentrated to the center. Secondly, no external forces are allowed to act upon them. Thirdly, an inertial frame of reference, which is absolutely at rest and relative to which all motion takes place, has to exist.

In the case of the solar system, the Sun (m_0) can be adopted as the origin of the coordinate system. The relative positions now refer to $\mathbf{r}_i \leftarrow \mathbf{r}_i - \mathbf{r}_0$ and the equations of motion become

$$\frac{d^2 \mathbf{r}_i}{dt^2} = -\gamma(m_0 + m_i) \frac{\mathbf{r}_i}{r_i^3} + \gamma \sum_{j=1, j \neq i}^N m_j \left(\frac{\mathbf{r}_{ij}}{r_{ij}^3} - \frac{\mathbf{r}_j}{r_j^3} \right), \quad (2.3)$$

where $i = 1, N$. The first term on the right-hand side describes the two-body motion of the central body m_0 and m_i . The second perturbative term represents the effect of the gravity of the other bodies on the Sun, i.e., it is a force that causes perturbations from Keplerian motion.

In a particular system where $N = 1$, Equation (2.3) describes the Keplerian two-body system (Figure 2.1). In this case, the equation of motion can be written as

$$\frac{d^2 \mathbf{r}}{dt^2} = -\mu \frac{\mathbf{r}}{r^3}, \quad (2.4)$$

where $\mu = \gamma(m_0 + m_1)$ and $\mathbf{r} = \mathbf{r}_1 - \mathbf{r}_0$. As there are three spatial coordinates in \mathbf{r} , this second-order differential equation requires six constants of integration for its complete solution.

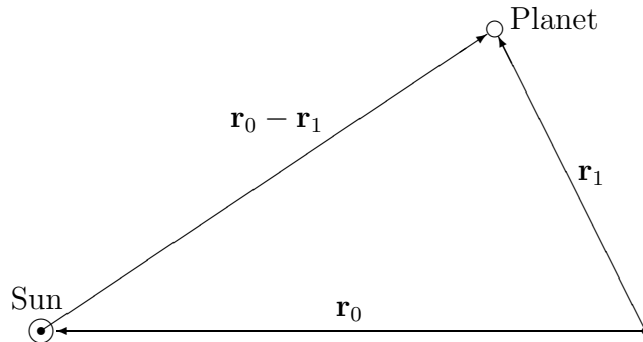


Figure 2.1: Illustration of the two-body problem.

The six constants of integration can be presented as Keplerian elements, Cartesian positions and velocities, Poincaré variables, etc. Because of the equivalence between these different sets, knowing one of them makes it possible to compute the others. In this thesis, all different sets of these constants are collectively referred to as the *orbital elements*.

The following two sections describe how the six constants of integration are solved with modern techniques. The sections are mainly based on the presentation of the methods given by Danby (1992).

2.2 f - and g -functions

The orbital reference system is a coordinate system with the origin at the attracting focus, the X -axis towards periapsis (the closest distance a body in orbit about a mass reaches), and the Y -axis in the direction for which the true anomaly

is 90° . The system is given by

$$X = r \cos \nu, \quad (2.5)$$

$$Y = r \sin \nu, \quad (2.6)$$

where r is the instantaneous radius and ν is the true anomaly. Time derivatives are given by

$$\dot{X} = -r\dot{\nu} \sin \nu + \dot{r} \cos \nu, \quad (2.7)$$

$$\dot{Y} = -r\dot{\nu} \cos \nu + \dot{r} \sin \nu. \quad (2.8)$$

Knowing six orbital elements, i.e., the position and velocity, at a given epoch, it is possible to obtain an exact mapping of these elements to an arbitrary epoch in the two-body approximation using the f - and g -functions, which are defined in the orbital reference frame:

$$\mathbf{r}(t) \equiv f(t, t_0)\mathbf{r}(t_0) + g(t, t_0)\mathbf{v}(t_0), \quad (2.9)$$

$$\mathbf{v}(t) \equiv \dot{f}(t, t_0)\mathbf{r}(t_0) + \dot{g}(t, t_0)\mathbf{v}(t_0). \quad (2.10)$$

By writing Equation (2.9) in component form

$$\begin{bmatrix} X(t) \\ Y(t) \end{bmatrix} = \begin{bmatrix} X(t_0) & \dot{X}(t_0) \\ Y(t_0) & \dot{Y}(t_0) \end{bmatrix} \begin{bmatrix} f \\ g \end{bmatrix}, \quad (2.11)$$

and solving f and g one gets

$$\begin{bmatrix} f \\ g \end{bmatrix} = \frac{1}{D} \begin{bmatrix} \dot{Y}(t_0) & -\dot{X}(t_0) \\ -Y(t_0) & X(t_0) \end{bmatrix} \begin{bmatrix} X(t) \\ Y(t) \end{bmatrix}, \quad (2.12)$$

where

$$D \equiv X(t_0)\dot{Y}(t_0) - Y(t_0)\dot{X}(t_0) = r_0^2\dot{\nu}_0. \quad (2.13)$$

Equation (2.12) can be written explicitly as

$$f = \frac{1}{D} [\dot{Y}(t_0)X(t) - \dot{X}(t_0)Y(t)] \quad (2.14)$$

$$= \frac{r}{p} [\cos(\nu - \nu_0) - 1] + 1, \quad (2.15)$$

$$g = \frac{1}{D} [-Y(t_0)X(t) + X(t_0)Y(t)] \quad (2.16)$$

$$= \frac{rr_0}{\sqrt{\mu p}} \sin(\nu - \nu_0), \quad (2.17)$$

where

$$p = r(1 + e \cos \nu), \quad (2.18)$$

and e is the eccentricity. By calculating the time derivatives of Equations (2.15) and (2.17) the two remaining functions can be found:

$$\begin{aligned} \dot{f} &= -\sqrt{\frac{\mu}{p^3}} [\sin(\nu - \nu_0) + e \sin \nu - e \sin \nu_0], \\ \dot{g} &= \frac{r_0}{p} [\cos(\nu - \nu_0) - 1] + 1. \end{aligned}$$

All orbit determination methods have to solve some version of Kepler's equation. Here, the computation of the difference between true anomalies ($\nu - \nu_0$) requires the equation to be solved to get the eccentric anomaly E . Instead of using traditional Keplerian variables and solving E from

$$M = E - e \sin E, \quad (2.19)$$

where M is the mean anomaly, the whole problem can be expressed in terms of universal variables. All different orbits can then be solved with a single set of formulas. In universal variables, the f - and g -functions, and their respective time derivatives, are written as

$$f = 1 - \left(\frac{\mu}{r_0}\right) s^2 c_2(\alpha s^2), \quad (2.20)$$

$$g = t - t_0 - \mu s^3 c_3(\alpha s^2), \quad (2.21)$$

$$\dot{f} = -\left(\frac{\mu}{r r_0}\right) s c_1(\alpha s^2), \quad (2.22)$$

$$\dot{g} = 1 - \left(\frac{\mu}{r}\right) s^2 c_2(\alpha s^2), \quad (2.23)$$

where the new independent variable s is defined as $dt = r ds$, and

$$\alpha = \frac{2\mu}{r} - \dot{\mathbf{r}}^2. \quad (2.24)$$

Terms c_1 , c_2 , and c_3 are so-called Stumpff functions:

$$c_k(x) = \frac{1}{k!} - \frac{x}{(k+2)!} + \frac{x^2}{(k+4)!} - \dots, \quad (2.25)$$

where $k=0,1,2,\dots$. Kepler's equation for solving s is written as

$$t - t_0 = r_0 s c_1(\alpha s^2) + r_0 \dot{r}_0 s^2 c_2(\alpha s^2) + \mu s^3 c_3(\alpha s^2), \quad (2.26)$$

and has to be solved iteratively as all versions of the equation.

2.3 p -iteration method for solving an orbit

The p -iteration method, also known as the method of Herrick and Liu, can be used to solve any kind of orbit assuming that two positions (\mathbf{r}_1 and \mathbf{r}_2) are known.

The only restriction is that it must be decided whether the orbital segment formed by \mathbf{r}_1 and \mathbf{r}_2 includes the attracting focus, or not.

Assume that ν_1 and ν_2 are the respective true anomalies for \mathbf{r}_1 and \mathbf{r}_2 . Then, in equations

$$C = \cos(\nu_2 - \nu_1) = \frac{\mathbf{r}_1 \cdot \mathbf{r}_2}{r_1 r_2} \quad (2.27)$$

$$\sin(\nu_2 - \nu_1) = Y \sqrt{1 - C^2} \quad (2.28)$$

$Y = 1$ if the orbital segment does not include the attracting focus, and $Y = -1$ if it does. Writing Equations (2.15) and (2.17) using Equations (2.27) and (2.28), the following equations are found:

$$C = \frac{\mathbf{r}_1 \cdot \mathbf{r}_2}{r_1 r_2}, \quad (2.29)$$

$$f = \frac{r_2}{p}(C - 1) + 1, \quad (2.30)$$

$$g = \frac{r_1 r_2}{\sqrt{\mu p}} Y \sqrt{1 - C^2}, \quad (2.31)$$

$$\mathbf{v}_1 = \frac{\mathbf{r}_2 - f \mathbf{r}_1}{g}, \quad (2.32)$$

where it is assumed that $g \neq 0$. If p is estimated, \mathbf{v}_1 can be computed. Having both \mathbf{r}_1 and \mathbf{v}_1 , a resulting position vector \mathbf{r}^* can be computed for the same epoch as \mathbf{r}_2 by using the f - and g -functions given by Equations (2.20) and (2.21). The cosine C^* of the angle between \mathbf{r}_1 and \mathbf{r}^* is

$$C^* = \frac{\mathbf{r}_1 \cdot \mathbf{r}^*}{r_1 r^*}, \quad (2.33)$$

which has to be compared with C from (2.27). The difference

$$D(p) = C - C^* \quad (2.34)$$

has to be driven to zero using an iterative process by changing the value of p . When zero difference has been reached, the correct \mathbf{v}_1 has been found. All six orbital parameters are thus known at the epoch of \mathbf{r}_1 , which means the orbit corresponding to that epoch has been found.

As stated above, the procedure can lead to a false solution depending on the value of Y . When \mathbf{r}^* is found from \mathbf{r}_1 and \mathbf{v}_1 , the value of the g function is computed. For a correct solution, the sign of g and Y should be the same. If it is not the case, then the correct solution can be found by changing the Y value and redoing the computations.

2.4 Criteria for acceptance

When fitting an orbit to observed positions, it is essential to have a criterion to estimate the goodness of the fit. A χ^2 - or rms-value (root mean square) can be

used for this purpose. χ^2 is defined as

$$\chi^2 = \frac{1}{N} \sum_{i=1}^N \frac{(X_i - x_i)^2}{\sigma_i^2}, \quad (2.35)$$

where N is the number of observations, X_i is the computed value, x_i is the observed value, and σ_i^2 is the corresponding standard deviation. Assuming that $\sigma_i^2 = \sigma^2$, Equation (2.35) can be written as

$$\chi^2 = \frac{1}{N\sigma^2} \sum_{i=1}^N (X_i - x_i)^2, \quad (2.36)$$

which simplifies the relation between χ^2 and rms:

$$\begin{aligned} \chi^2 &= \frac{1}{\sigma^2} \frac{1}{N} \sum_{i=1}^N (X_i - x_i)^2 \\ &= \frac{1}{\sigma^2} \left(\sqrt{\frac{1}{N} \sum_{i=1}^N (X_i - x_i)^2} \right)^2 \\ &= \frac{1}{\sigma^2} \text{rms}^2. \end{aligned} \quad (2.37)$$

For statistical inversion methods it can, for instance, be required that

$$\Delta\chi^2 = \chi^2 - \chi_{min}^2 \quad (2.38)$$

is smaller than a given threshold. χ_{min}^2 may correspond to, e.g., the least-squares orbit, or the best fit sample orbit.

Chapter 3

Algorithm for identification at discovery

The simplest way to perform the identification is to combine two observational data sets and make the inversion from observations to orbital elements with statistical ranging (Virtanen et al. 2003). If even a single solution is found, it shows that the observational sets can be linked. In the case of larger sets of observations leading to multiple solutions, the final decision has to be made either by further observations, or by using a metric describing the probability of an identification. The use of other inversion methods looking for a single orbit is somewhat questionable since they might not find a solution even if the identification would be correct. The direct method has, however, a major disadvantage, namely the huge computational load.

A realistic identification problem consists of two observational sets, A and B . It is assumed that all identifications have been done correctly within each set. Assuming that the number of objects in each set is $n_A \approx n_B \approx 10^3$, then the number of possible identifications is $n_A n_B \approx 10^6$. As the maximum number of correct identifications is the smaller one of n_A and n_B , it is clear that in this scenario more than 99.9% of the possible identifications are false. Thus, it is a waste of resources to try to make the inversion for all possible matches. The identification is most efficiently performed by filtering out unlikely pairs in a multistep process.

All steps in the identification procedure described here use statistical ranging as the inversion method. Therefore, it is shortly described before presenting the methods used in the identification process. Several methods have been developed and tested for the first step, as its efficiency is crucial for the whole identification algorithm. Consequently, Section 3.2 (particularly Section 3.2.3) is the key section of the thesis.

3.1 Statistical orbital ranging

The technique of statistical orbital ranging is currently being modernised by the asteroid orbit computation team (K. Muinonen, J. Virtanen and M. Granvik) from the version presented by Virtanen et al. (2001). Instead of using Keplerian elements during the inversion, Cartesian positions and velocities are used throughout the process. The implementation of the revised method is written in Fortran 95 using an object oriented programming paradigm.

The probability density of orbital elements is examined using Monte Carlo selection of orbits in orbital element space in the following way:

- Two observations are chosen (usually the first and the last), and angular deviations in R.A. and Dec. are introduced.
- Topocentric ranges (distances) are assumed corresponding to the observation dates. In other words, two positions equalling six constants of integration are known.
- A trial orbit is computed using the p -iteration method described in Section 2.3 and is compared to all observations. If the trial orbit fits the observations to predefined accuracy (defined as a $\Delta\chi^2$ -threshold and maximum sky-plane residuals), it is added to the sample of possible orbits.

The initial topocentric range intervals are determined manually using an educated guess in the basic version of statistical ranging, whereas in the automated version the topocentric range intervals are further improved using the $3\text{-}\sigma$ cutoff values of the range probability density. By increasing the number of generated sample orbits ($10 \rightarrow 200 \rightarrow n$), an unbiased phase space region of possible orbits is found. Besides the $\Delta\chi^2$ -threshold and maximum sky-plane residuals, the number of trials and solutions, as well as the initial topocentric range intervals, are the key parameters of statistical ranging.

A probability density value, which describes how well the orbit explains the observations, is associated with every orbit in the discrete orbit distribution. Neglecting the probabilities, the distribution merely shows the extent of different orbital solutions assuming a predefined observational error. A priori knowledge on the orbital elements, mainly the semimajor axis a , eccentricity e and inclination i , may be used to narrow down the orbit distribution, if necessary.

Ephemeris are generated by transforming the orbital elements of every orbit to a position on the celestial sphere at a given epoch. The result is thus a discrete ephemeris distribution, which is interpreted as a continuous ephemeris probability density as the number of orbits approaches infinity (practically some thousands).

3.2 Removing the bulk of impossible candidates

The first step in the filtering process has a clear objective: it should remove the bulk of impossible candidates. The algorithm chosen for this purpose needs to fulfil two main criteria, namely speed and reliability. Speed is important because of the vast number of possible identifications to scan through. At the same time it is essential to have a reliable routine which does not exclude correct identifications. It is certainly better to allow a few more erroneous identifications, than to remove the only correct one.

There are essentially four quantities that can be compared: astrometric positions (i.e. observations), magnitudes, ephemerides, or different orbital elements. Magnitudes are difficult to use in short-term identification, because knowledge of the physical properties of the targets is naturally very modest close to discovery. The three remaining quantities can be used in the following combinations: ephemerides vs. observations, elements vs. elements, or ephemerides vs. ephemerides. The following three sections describe a method for each of these combinations in the chronological order they were examined.

A common problem among Monte Carlo methods is how to efficiently examine the large output of data, here manifested particularly in the *asymmetric ephemeris comparison* technique. It turned out, that at least for some cases the huge computational load can be overcome by simplifying the data as much as possible. The key idea is to compress real number arrays to integers, and hence be able to make data comparisons more efficiently. The new technique, termed *address comparison*, is described in connection to the *phase space address comparison* method, but it is also a main constituent of the best one of the examined first-step algorithms, i.e., the *symmetric ephemeris address comparison*.

3.2.1 Asymmetric ephemeris comparison

Using observations of an object in set A , a distribution of ephemerides is generated corresponding to the first and the last epoch of an object in set B . If neither the first position (R.A. and Dec.), nor the positional difference computed between the first and the last position (Figure 3.1) fits the observations in set B to predefined accuracy, the pair is excluded from possible identifications.

Asymmetric ephemeris comparison is efficient in finding the correct identifications. As long as the maximum differences between the coordinates of approved identifications are kept small, it seems like the ratio between false positives and correct identifications could go down to approximately one, requiring that all correct ones have been found. For a first-step algorithm this is excellent.

Difficulties with the asymmetric approach are a huge computational load due

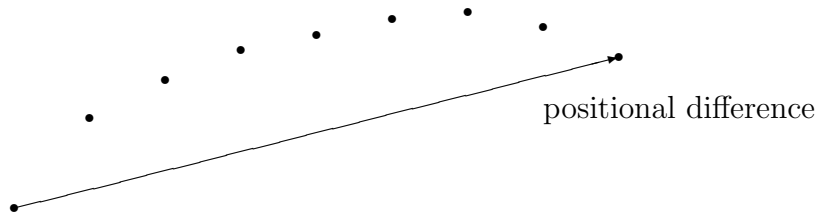


Figure 3.1: The geometric definition of the positional difference between the end-points of the nonlinear motion. Filled circles are observations.

to computations of the distances between observations and ephemerides, and the asymmetry in the treatment of observational sets (hence the term *asymmetric*). The computational load arises from the computations of distances between observations and theoretical positions, and between observed and theoretical positional differences. Since observations are given in spherical coordinates, the computation of distances between positions requires the use of trigonometric functions, which in turn is computationally heavy. Moreover, the distances have to be measured between every ephemeris and every observation. Consider for instance the following example: observations of every object in set A ($n_A \approx 10^3$) are transformed into orbit distributions containing $n_{orbit} = 2 \times 10^3$ individual orbits. All these orbits ($N_{orbit} = n_A n_{orbit} \approx 2 \times 10^6$) are then transformed into ephemerides at the same epochs as the observations in set B ($n_B \approx 10^3$). Put together, the two distances (between positions, and between positional differences) have to be computed for $N_{orbit} n_B \approx 2 \times 10^9$ separate cases. When the distances have been computed, they still need to be analysed, i.e., the shortest distances need to be found.

An obvious question rising from the asymmetric treatment of observational sets, is whether this method produces similar results if the sets are switched? The answer is clearly no. Even if the parameters would be adjusted to find all correct pairs, the number of false positives will always depend on the order of the observational sets. To avoid subjective interpretation of the data, it is advisable to treat both observational sets equally.

3.2.2 Phase space address comparison

Observations in both sets are inverted to orbital elements, e.g., Keplerian elements or Poincaré variables. The epoch of the elements is arbitrary, but it is most convenient to choose the mean epoch of the combined observation set. As both sets of observations have been treated similarly while inverted to orbital el-

elements, and integrated to the same epoch, they can be compared with each other to find similar values indicating possible identifications. There are two ways of comparing the distributions; either by measuring distances between sets orbital elements, or by binning the distributions and searching for bins populated by elements inverted from two different observational sets.

In the first method, a possible identification is accepted if the distance between orbital elements is smaller than a given cutoff value. There are two problems with this approach. First, measuring the distances between two sets of orbital elements requires some kind of metric. This could be solved by using a metric resembling those used for determining asteroid families by, for instance, Nesvorný et al. (2002):

$$d_p = na_p \sqrt{C_a(\delta a_p/a_p)^2 + C_e(\delta e_p)^2 + C_i(\delta \sin i_p)^2}, \quad (3.1)$$

where p denotes proper elements, d_p is the distance in the (a_p, e_p, i_p) -space, na_p is the heliocentric velocity of an asteroid on a circular orbit having semimajor axis a_p , $\delta a_p = |a_p^{(1)} - a_p^{(2)}|$, $\delta e_p = |e_p^{(1)} - e_p^{(2)}|$, and, $\delta \sin i_p = |\sin i_p^{(1)} - \sin i_p^{(2)}|$. The indices (1) and (2) denote the two bodies under consideration. C_a , C_e , and C_i are constants, e.g., 5/4, 2, and 2, respectively.

Since the elements hardly change during short periods of days or months, osculating elements can be used instead of proper elements. As the computation of proper elements requires the osculating elements to be very well known, the use of proper elements would, in fact, be an impossible solution to the short-term identification problem. For Keplerian elements, the suitable metric could thus be the following:

$$d = na \sqrt{C_a(\delta a/a)^2 + C_e(\delta e)^2 + C_i(\delta \sin i)^2}, \quad (3.2)$$

where d is the distance in the (a, e, i) -space, na is the heliocentric velocity of an asteroid on a circular orbit having semimajor axis a , $\delta a = |a^{(1)} - a^{(2)}|$, $\delta e = |e^{(1)} - e^{(2)}|$, and, $\delta \sin i = |\sin i^{(1)} - \sin i^{(2)}|$. The indices (1) and (2) now denote the two orbits under consideration, and C_a , C_e , and C_i are the same constants as above.

The other problem is, once again, a heavy computational load. Despite that the computation of the evaluation function is straightforward, the computational load is heavy due to the need to measure the distance between every orbit inverted from observations in set A and every orbit inverted from observations in set B . Using the numbers given for the example in connection to the asymmetric ephemeris comparison above, approximately 4×10^{12} of these comparisons are needed.

The second method, binning the orbital elements, is more promising. The idea is to put a multidimensional grid in the orbital element space and fill the

bins between the gridpoints with orbits. In other words, each bin is related to similar orbits. The difference between orbits in a bin depends on the bin size. It is a very efficient way of categorising the orbits; using the above example, only 4×10^6 orbits have to be put in the correct bins. A possible identification is approved, if orbits inverted from both observational sets are found in the same bin.

The problem with binning is the small bin size compared to the large volume of the interesting orbital element space. Combined with the relatively small amount of objects, this leads to a binned array mostly containing empty bins. Consider, for instance, the binning given in Table 3.1. Assuming the same amount of objects and orbits as above, the maximum amount of populated bins is 4×10^6 out of approximately 1.7×10^{12} . In other words, there are at least some 400,000 empty bins per populated bin.

Element	Min. value	Max. value	Bin size	No. of bins
a	0 AU	10 AU	0.5 AU	20
e	0	1	0.1	10
i	0°	180°	1°	180
Ω	0°	360°	1°	360
ω	0°	360°	1°	360
M	0°	360°	1°	360

Table 3.1: Parameters that have been found suitable for phase space comparison using Keplerian elements.

When searching through the bins to find possible identifications, the time is thus mostly spent checking empty bins, which is naturally not efficient. Instead of using the whole map, or multidimensional array, one can write an address to each bin and just compare the addresses that are occupied with orbits.

The phase space comparison method, and particularly the address comparison technique, has originally been developed by K. Muinonen. For this thesis, the technique was optimised and ported to Fortran 95. Moreover, the transformation algorithm described below was generalised to allow arrays of any size and values.

In practice, the address is an integer $i \in \mathbb{N}_+$, transformed from an array of elements $p \in \mathbb{R}^6$ using a transformation algorithm f , i.e., $i = f(p, \dots)$. The transformation algorithm f essentially does the same as a basic binning algorithm. But instead of returning the coordinates of a bin in multidimensional space (a bi-product of the algorithm), it transforms the coordinates to a single integer i . The integer can be thought of as a coordinate of an element in a long, winding one-dimensional array perfectly filling the multidimensional space. Each element

in this array corresponds to a separate bin in the original multidimensional bin-network. Besides p , the essential input values for the transformation algorithm f are the boundary values of the multidimensional space and the bin sizes. An implementation of the technique in Fortran 95 is found in Appendix A.

Since the observations of an object are inverted to an orbit distribution, and every orbit is transformed to a value i , each object will get a one-dimensional array containing the i -values. Potential identifications are searched by comparing the i -arrays of objects in set A with the i -arrays of objects in set B . The search can move to the next possible identification as soon as a single pair of equal addresses, or integers, are found.

Using the example, a maximum of 4×10^{12} comparisons between addresses would need to be performed. Although the magnitude of the number is the same as when computing differences between orbits, there is one big advantage. When dealing with an array of integers, the search algorithm can be optimised easier, than when searching a multidimensional array. By sorting the i -values in ascending order, a *binary search* algorithm can be used for the search of similar elements, which significantly accelerates the comparison algorithm. The suitable sorting algorithm for this implementation of the comparison algorithm is *insertion sort*, which is very efficient for almost sorted arrays. Short descriptions of these algorithms are given in Appendix B.

The use of the address comparison technique makes the phase space comparison method much faster than, e.g., the asymmetric ephemeris comparison method. As a rule of thumb, it is approximately ten times faster for the same data set.

The main weakness of a method using orbital elements for short-term identification arises from the well-known characteristics of widely spread orbital element distributions for objects with short observational arcs or small numbers of observations (e.g. Virtanen et al. 2001, Howell et al. 2003). Wide distributions lead to a situation, where an object can be linked with almost any other object, leading to an unacceptably high number of false positives. It turns out, that most efficiently the number of false positives are reduced by taking into account the locations of the objects. In Keplerian elements it means comparing the values of the inclination i , the longitude of the ascending node Ω , the argument of perihelion ω , and the mean anomaly M .

3.2.3 Symmetric ephemeris address comparison

By combining the strengths of the asymmetric ephemeris comparison (accuracy) and orbital element address comparison (speed), an efficient *symmetric ephemeris address comparison* method is found. Even better, also the weaknesses of these

two earlier methods seem to vanish as their better features are utilised.

The idea is to generate ephemeris distributions (R.A. and Dec.) and position differences (Δ R.A. and Δ Dec.) for objects in both sets at the same epoch (hence the term *symmetric*), and then find out whether any objects in different sets have similar values, which would indicate a possible identification. The choice of epoch can be optimised, but the mid-epoch is a good approximation to start from, based on the knowledge that the ephemeris uncertainty grows with increasing time since last observation (Muinonen et al. 1994). Position differences are found by computing the difference between two ephemerides generated at epochs separated by twelve hours. The second ephemeris is produced by integrating an ephemeris distribution, which is generated at the first epoch, to the second epoch, thus mapping the nonlinear motion rigorously.

The search for similarities among the two ephemeris is done using the address comparison technique originally developed for the phase space address comparison method.

3.3 Sieving possible linkings

The second stage in the filtering process is a kind of a preparation for the final step. During the final filtering, an extensive orbit distribution is produced which requires a huge number of trial orbits. However, if even a single orbit can not be found in the inversion process, it means that valuable cpu-time is wasted. The number of pairs, for which orbits can probably not be found, should therefore be minimised before entering the last phase.

By using four observations (e.g., the two first and the two last observations of the combined set) available for a possible identification, one orbit is generated by using no more than a given number of trials. If no orbit is found, the pair at issue is excluded from the list of possible identifications.

The observations belonging to pairs of initial objects, which still have not been excluded, may or may not belong to a single object. At least one orbit could be found that ties the used observations to a single object. Since all observations are not necessarily used in this step, there will be some false positives slipping through.

3.4 Finding potential identifications

During the last step of the identification algorithm, all observations related to a potential identification are used to generate an extensive orbit distribution. The simple fact that this inversion from observations to orbits is possible assuming

given observational errors, is the sensible end for this algorithm. From this point forward, the rational continuation of the identification process would be to make observations using ephemerides computed for possible identifications. If an object is observed moving as predicted by the ephemerides, it can be considered an identification. On the other hand, if an object can not be found, the pair is obviously a false positive and can thus be ruled out.

It is, however, possible to go further using only the knowledge given by the original observations, and orbits inverted from them. Nevertheless, it can not be stressed too much, that all further results are uncertain; all identifications fit the observations to the accuracy given by the assumed observational error. Being aware of this limitation, one can consider, for instance, a case with two possible identifications, where letters denote observational sets, and numbers are different objects within a set:

$$\begin{aligned} A1 &\longleftrightarrow B1 \\ A1 &\longleftrightarrow B2 \end{aligned}$$

As $A1$ can not be linked to both $B1$ and $B2$, one can compute different quantities describing the quality, or probability, of the fit based on the orbit distribution. For instance a χ^2 -value could be used to rule out the first of two possible identifications:

$$\begin{aligned} A1 &\longleftrightarrow B1 & \chi^2 = 0.8 \\ A1 &\longleftrightarrow B2 & \chi^2 = 0.2 \end{aligned}$$

Usually the best fit corresponds to the correct identification, but it should not be taken for granted due to observational errors. The complexity of the problem can be seen in the, more or less theoretical, example given below.

$$\begin{aligned} A1 &\longleftrightarrow B1 & \chi^2 = 0.8 \\ A1 &\longleftrightarrow B2 & \chi^2 = 0.2 \\ A1 &\longleftrightarrow B3 & \chi^2 = 0.4 \\ A2 &\longleftrightarrow B2 & \chi^2 = 0.1 \\ A3 &\longleftrightarrow B1 & \chi^2 = 0.6 \\ A3 &\longleftrightarrow B2 & \chi^2 = 0.5 \\ A3 &\longleftrightarrow B3 & \chi^2 = 0.3 \end{aligned}$$

Excluding the χ^2 -values and trying to find the candidate pairs which have to be taken into account while looking for the most probable identification combining $A1$ with an object from set B , it turns out one must actually check every possible identification in this set.

Moreover, if the χ^2 -values are taken into account, it turns out that the solution is different depending on the selection method that is used. For instance, if the criterion is to find the smallest possible χ^2 -value for each individual identification by progressing step-wise through the list of possible identifications, the solution is $A1 \leftrightarrow B1$, $A2 \leftrightarrow B2$, and $A3 \leftrightarrow B3$. For some cases, the results may even depend on how the route through the possible identifications is chosen, i.e. one might end up with different results when beginning from the top of the list of possible identifications, compared to doing it the other way round and start from the bottom.

On the other hand, if the criterion is to minimise the total sum of χ^2 -values for all identifications while simultaneously maximising the number of identifications, the solution is $A1 \leftrightarrow B3$, $A2 \leftrightarrow B2$, and $A3 \leftrightarrow B1$. It is obviously questionable, whether the requirement to find as many identifications as possible is clever at all.

As is probably already evident, there is only one smart thing to do at this stage: grab a telescope, and either accept or reject possible identifications through further observations.

Chapter 4

Simulated observations

The goal is to have sets of observations of NEOs and MBOs resembling those produced by real-life asteroid search programs. They are observing the same part of the sky twice during a single night with an interval of approximately one hour. Instead of returning to a certain area every fourth or fifth day, like real search programs do, the area at issue should be covered during eight subsequent nights. A continuous set of observations is needed to get an idea of how the number of false positives produced by the identification algorithm evolves as the observational time arc between subsets of observations increases.

There are a lot of data available¹ that need to be scanned with a proper identification algorithm. Why are they not utilised in this study? First of all they can not be used for estimating the probability of success since the correct identifications are not known *a priori*. Secondly, it is difficult to control the epochs of real observational data, which makes it impossible to analyse the impact of other variables such as variations in limiting magnitude. Thirdly, they might contain false detections, e.g., cosmic rays. Fourthly, it is not granted that they contain a proper sample of the populations, particularly when reaching deeper magnitudes. Fifthly, observational errors are not available, which makes it hard to estimate the intrinsic errors of the identification algorithm. Sixthly, by using simulated observations it is possible to make a decision between a two-body approximation and n-body integration. If using the two-body approximation while generating observations, it is possible to use the same fast approximation in the identification algorithm. This will significantly reduce the computational load without producing uncontrollable errors.

¹E.g. LONEOS unidentified objects archives available via <http://asteroid.lowell.edu/>.

4.1 ASurv simulation software

The current asteroid survey simulation software called **ASurv** (Muinonen 2003) has been updated from that used by Bowell & Muinonen (1994), Muinonen (1998) and Tedesco et al. (2000), and for the thesis it was translated to Fortran 95. A debiased population of NEOs and MBOs is generated using the latest knowledge of the observational biases, the overall orbital element distribution (Jedicke et al. 2003), and the orbital elements of the known objects provided by Bowell et al. (1994) in the `astorb.dat` file.

The software allows the simultaneous simulation of several different surveys (e.g. visual or infrared) from various locations. It uses the ephemerides provided by the Jet Propulsion Laboratory (Standish 1998) in the `de405.dat` file to determine the position and motion of the Earth with respect to the Sun at a given time.

The asteroid survey simulation is essentially a Monte Carlo simulation. The orbital elements, diameter, and geometric albedo are randomly generated for a sample asteroid using knowledge on the assumed distributions of orbital elements and absolute magnitudes in the process. Positions are then computed in the two-body approximation for the given survey dates. Finally, a decision is made whether the sample asteroid is observable or not, based on the specified region(s) and limiting magnitude(s).

It should be stressed that instrument-dependent parameters, such as the trailing loss², are not taken into account while deciding whether an object is observable or not. Instead, one could think the observations are made with a perfect instrument, the integration time of which approaches zero.

4.2 Generating simulated observations

Originally the intention was to perform the analysis for three different limiting magnitudes V ; 20^m , 22^m and 24^m . These magnitudes roughly correspond to the surveys at present, in the near future, and in a more distant future, respectively. As the computational power is limited, the number of observed objects need to be kept reasonable, say to about one thousand. Knowing that the number of objects grows exponentially with increasing absolute magnitude H (see Figure 1.4), the observed area must decrease proportionally. Otherwise, the number of observations could not be kept small enough. While generating simulated observations, it became clear that suitable areas for each limiting magnitude are $8^\circ \times 8^\circ$, $3^\circ \times 3^\circ$, and $1^\circ \times 1^\circ$, respectively. Unfortunately, it later turned out

²Loss of magnitude due to the fast movement of an object on the CCD-chip.

that the area corresponding to limiting magnitude 24^m is too small in the sense that every linkable object has moved beyond the area after a few days. In other words, the observational sets are useless for identification studies after a couple of nights since there are only false linkings available. Using a larger area would result in computational overload, thus stating clearly that going this deep in magnitude is a dead end for the time being. In a larger sense, this could be seen as a statement of the problem's size. Future projects, e.g., Pan-STARRS, will scan the whole night-sky in four days to the limiting magnitude at issue, which means that several hundreds of thousands of objects must be routinely identified every night.

As $V = 24^m$ proved to be almost impossible for the time being, a decision was made to change the limiting magnitudes to 18^m , 20^m and 22^m . These new values roughly correspond to advanced amateurs, professional surveys, and professional surveys in the near future. The area suitable for $V = 18^m$ is $25^\circ \times 25^\circ$.

In order to use the simulation software efficiently, one should not go deeper in absolute magnitude H than necessary. On the other hand, the simulation must reach deep enough to produce a realistic outcome. The correct H is found when the number of observed objects does not increase when the absolute magnitude interval is increased and the limiting, or apparent, magnitude stays constant. This H -limit has to be determined separately for every V to be used.

A suitable limit for the absolute magnitude is practically found by running the simulation in a loop and adding a small dH to H during every loop. After the observations are generated during a loop, they are counted. Because of the statistical nature of the simulated set of observations, the number of sample asteroids must be larger than the realistic amount of asteroids corresponding to the absolute magnitude at issue. To get an efficient routine and realistic result, the number of sample asteroids is therefore equal to the assumed number of asteroids below a certain absolute magnitude multiplied with a constant. The value of the constant affects the smoothness of the curve, and since the number of observed objects is renormalised back to a realistic level, its choice is more or less arbitrary. The constant is chosen to be 100 for NEOs, and 10 for MBOs. The whole sky, i.e., not only the dark sky, is used as the observational window. Observational losses, e.g., the brightness of the background sky on small solar elongations, are not taken into account.

As seen in Figures 4.1 and 4.2, the location of the steep slope is roughly proportional to the limiting magnitude. In the case of MBOs, finding the suitable H -limit is quite straightforward. The limits were chosen to be 17^m , 19^m , and 21^m for limiting magnitudes 18^m , 20^m , and 22^m , respectively.

NEOs are more problematic. As can be seen in Figure 4.2, the number of NEOs brighter than a given limiting magnitude does not necessarily flatten out as

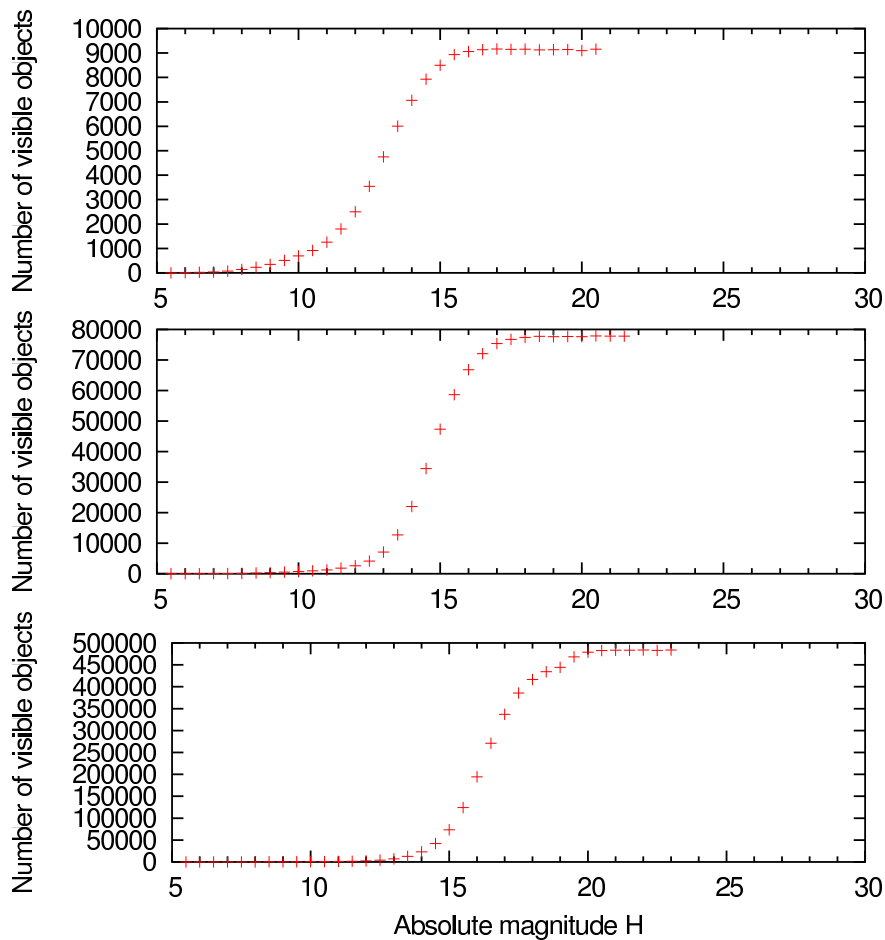


Figure 4.1: Number of MBOs brighter than a given limiting magnitude at a fixed date as a function of the absolute magnitude H . The limiting magnitudes are (from the top) 18^m , 20^m , and 22^m . These numbers were generated for a geocentric observer without considering observational losses due to, e.g., the varying brightness of the background sky.

a function of the absolute magnitude. This can be explained by two main factors, namely the geocentric distance and the characteristics of the NEO population. As the minimum geocentric distance for NEOs can be very small, there are loads of faint bodies in terms of H that come sufficiently close to Earth to be detected. In other words, it is a question of distance; if one of these faint bodies would be moved from near-Earth space to the main belt, it would be too faint to be observed. As these faint NEOs must reach small geocentric distances to be visible, their apparent motion will usually be quite large. Now, if one considers a real telescope, these fast and faint objects could probably not be observed due to the trailing loss. Furthermore, motions sticking out from the crowd do not pose any problems for identification and are thus not interesting for this study.

Secondly, the population characteristics also play a role. If one considers the different slope parameters for the differential H distribution of NEOs given by

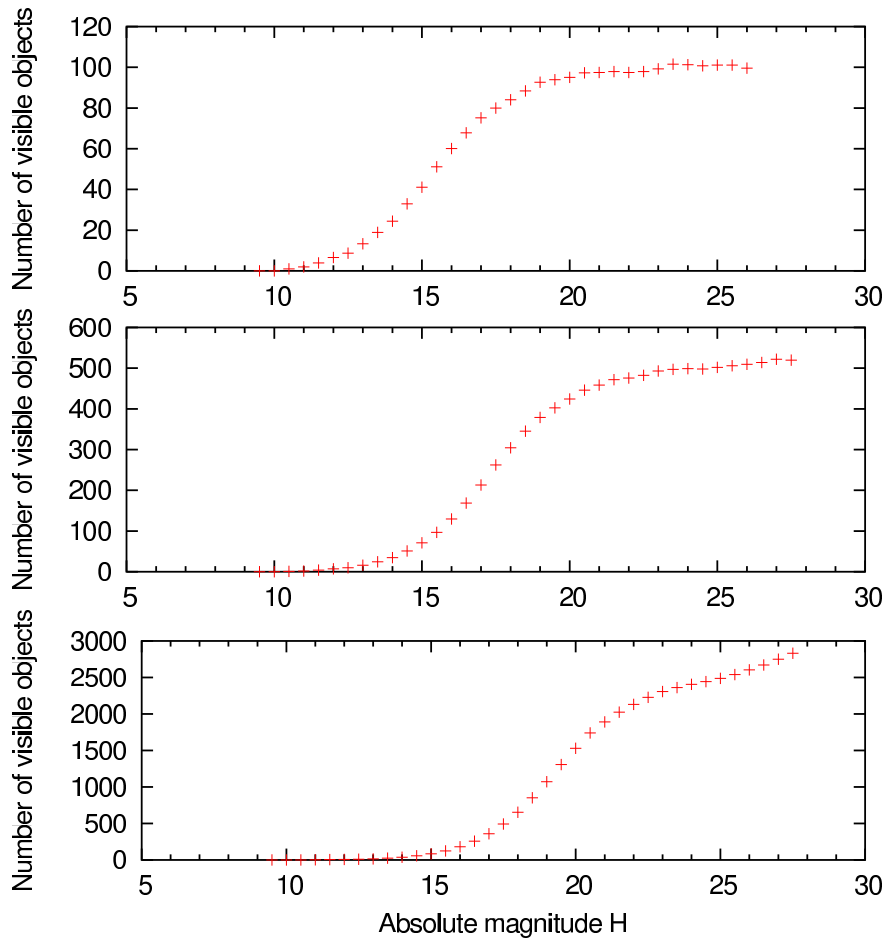


Figure 4.2: Same as Figure 4.1, but for NEOs.

Jedicke et al. (2003), it is obvious that the number of NEOs grows more rapidly for $H > 24^m$ than when $H < 24^m$ (Figure 1.4). This effect can also be seen in the lowest plot in Figure 4.2, at $H \gtrsim 24^m$. As the number of NEOs compared to MBOs is just of the order of one per cent, going a few magnitudes deeper does not substantially change the situation. A decision is therefore made to set a hard limit at $H = 25^m$ for NEOs for all limiting magnitudes. The smallest NEOs used in this study are thus comparable in size (some tens of metres) to the asteroid that exploded in the atmosphere above Tunguska River in 1908 (e.g. Morrison et al. 2003).

As discoveries are usually made at or near opposition, the direction of opposition at the discovery date is chosen to be the center of the observed area. A search program is then mimicked by generating two nightly sets of simulated observations of the same area during eight sequential nights. In order to keep things simple, the virtual observer resides in the center of Earth thus producing geocentric observations. The same area is reached by turning the virtual telescope in the opposite direction (clockwise, if seen from the north ecliptic pole), but at the same angular velocity as the Earth is orbiting the Sun. The angular

velocity is

$$\frac{360^\circ}{P_{Earth}} = \frac{360^\circ}{365.25 \text{ d}} \approx 0.9856^\circ \text{ d}^{-1} \quad (4.1)$$

Obviously, if one considers a volume of space close to Earth it will not be covered on sequential nights, as the distance to the perfectly covered area is infinite.

The time interval between two nightly sets is approximately one hour (exactly 0.04 days), which is, as already mentioned, roughly the same as for currently running search programs.

Finally, single observations in the combined nightly sets were sorted out, because there is no way to decide whether they belong to moving objects, or not. A Gaussian error ($\sigma = 0.5''$) is also added to the observations. Now the set of simulated observations is ready to be used for the study of identification problems. But before going into the main topic, a few interesting things about the generated observations should be brought up.

4.3 Observational properties of the simulated asteroid population

4.3.1 Verification

Figures 4.1 and 4.2 show the maximum number of visible objects at a certain limiting magnitude for the whole sky. If a similar curve would be made for a smaller part of the sky, say for a few square degrees, it could be compared to the number of real detections for a similar area as an independent verification of the simulated population.

4.3.2 Statistical opposition effect

A statistical opposition effect is manifested as a decreasing number of objects observed as the telescope is pointed further away from the direction of opposition. Since the opposition effect occurs in a relatively narrow area centered in the direction of opposition, it should be more evident for smaller observational windows. In Figure 4.3 this can be seen as a pronounced opposition effect for higher limiting magnitudes, which in this case correspond to smaller observation areas. The values on the x -axis can be converted to degrees by multiplying with the angular velocity given by equation 4.1. The discovery day naturally equals 0° , while the phase angle is approximately 7° a week later.

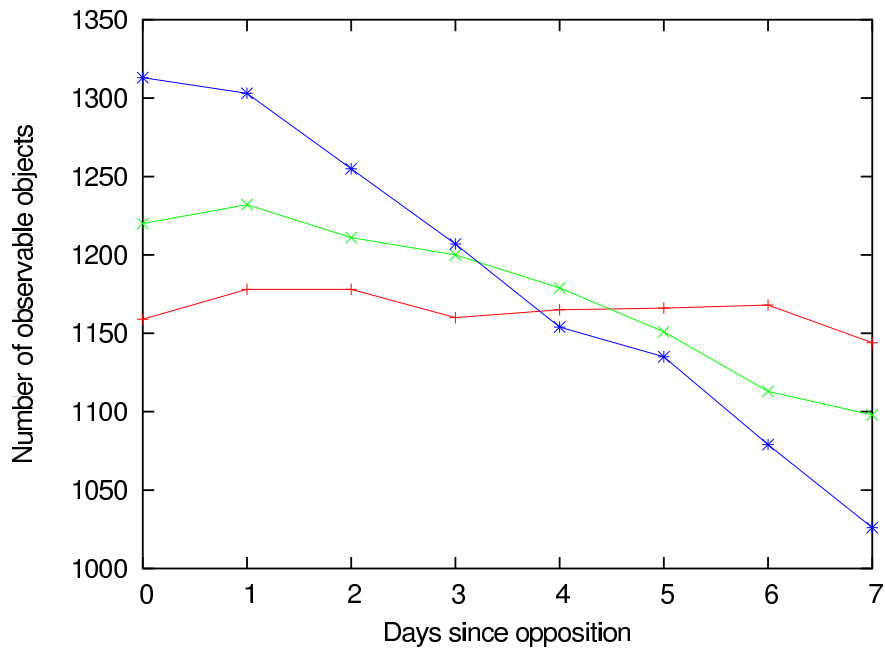


Figure 4.3: The number of observable objects, as a function of days since opposition, at different limiting magnitudes. Note that the key parameter is not the limiting magnitude itself, but the size of the observational area corresponding to each limiting magnitude. A line is used to connect the values between sequential nights to make the general trend more clearly visible. The symbols correspond to the following limiting magnitudes and observational areas: (+): $V = 18^m / 25^\circ \times 25^\circ$, (x): $V = 20^m / 8^\circ \times 8^\circ$, and (*): $V = 22^m / 3^\circ \times 3^\circ$.

4.3.3 Positions and motions

Figures 4.4, 4.5 and 4.6 show the discovery observations of the objects at limiting magnitudes 18^m , 20^m , and 22^m , respectively. The center of each image is in the direction of opposition.

Figure 4.7 shows the position differences of the objects during one hour on the discovery night. Both NEOs and MBOs are displayed in these plots, and the limiting magnitudes, V , are 18^m , 20^m and 22^m .

For discovery and identification purposes it is interesting to see how fast the objects are spreading on the celestial sphere. Figures 4.8, 4.9 and 4.10 show the expansion of approximately one thousand MBOs during one week. Figures 4.11, 4.12 and 4.13 show the same for NEOs. The squares mark the outer edges of the sample at discovery, and the circles map the situation one week later. The observational windows at discovery are $25^\circ \times 25^\circ$, $8^\circ \times 8^\circ$ and $3^\circ \times 3^\circ$, and the respective limiting magnitudes are 18^m , 20^m and 22^m . While generating the NEOs, the number of sample asteroids was equal to the realistic number of asteroids corresponding to the absolute magnitude used (25^m for all NEO figures) multiplied by 300 to get a big enough sample. Thus, the NEO figures are not

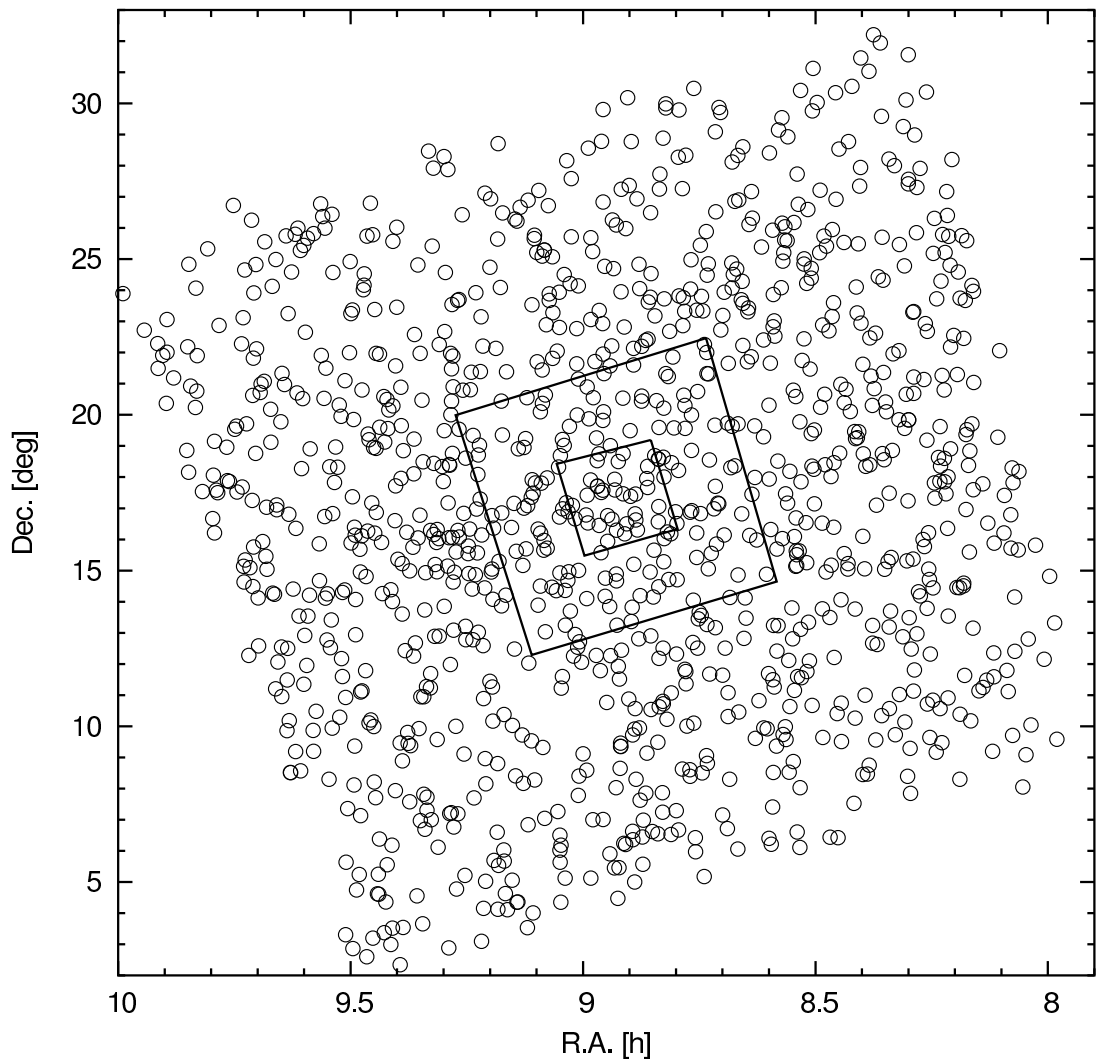


Figure 4.4: Discovery observations of 1159 objects down to $V = 18^m$ in the direction of opposition. The squares mark the regions covered at limiting magnitudes 20^m (large, see Figure 4.5) and 22^m (small, see Figure 4.6).

showing a realistic number of observations, but merely a realistic scattering of objects with time.

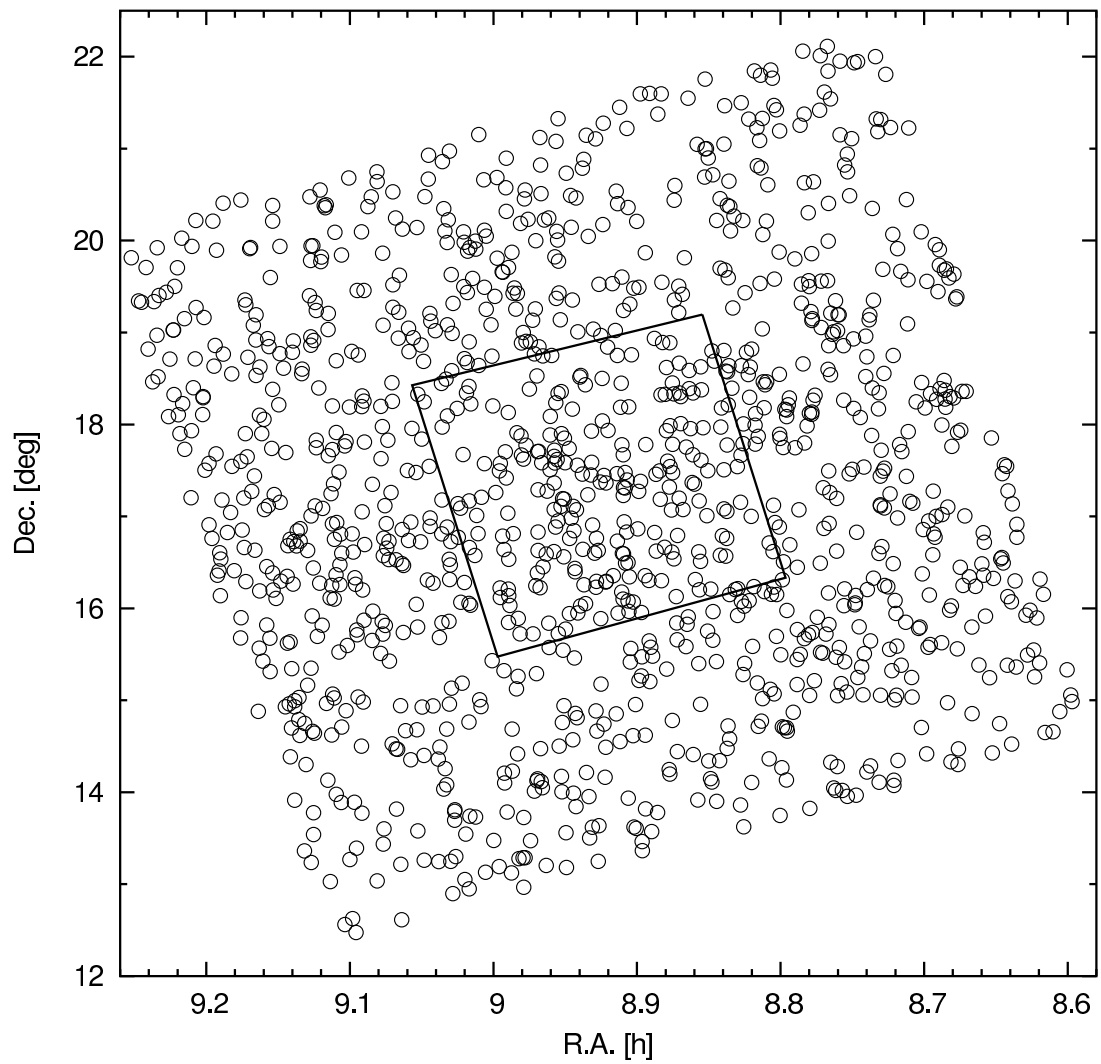


Figure 4.5: Discovery observations of 1220 objects down to $V = 20^m$ in the direction of opposition. The square marks the region covered at limiting magnitude 22^m (see Figure 4.6).

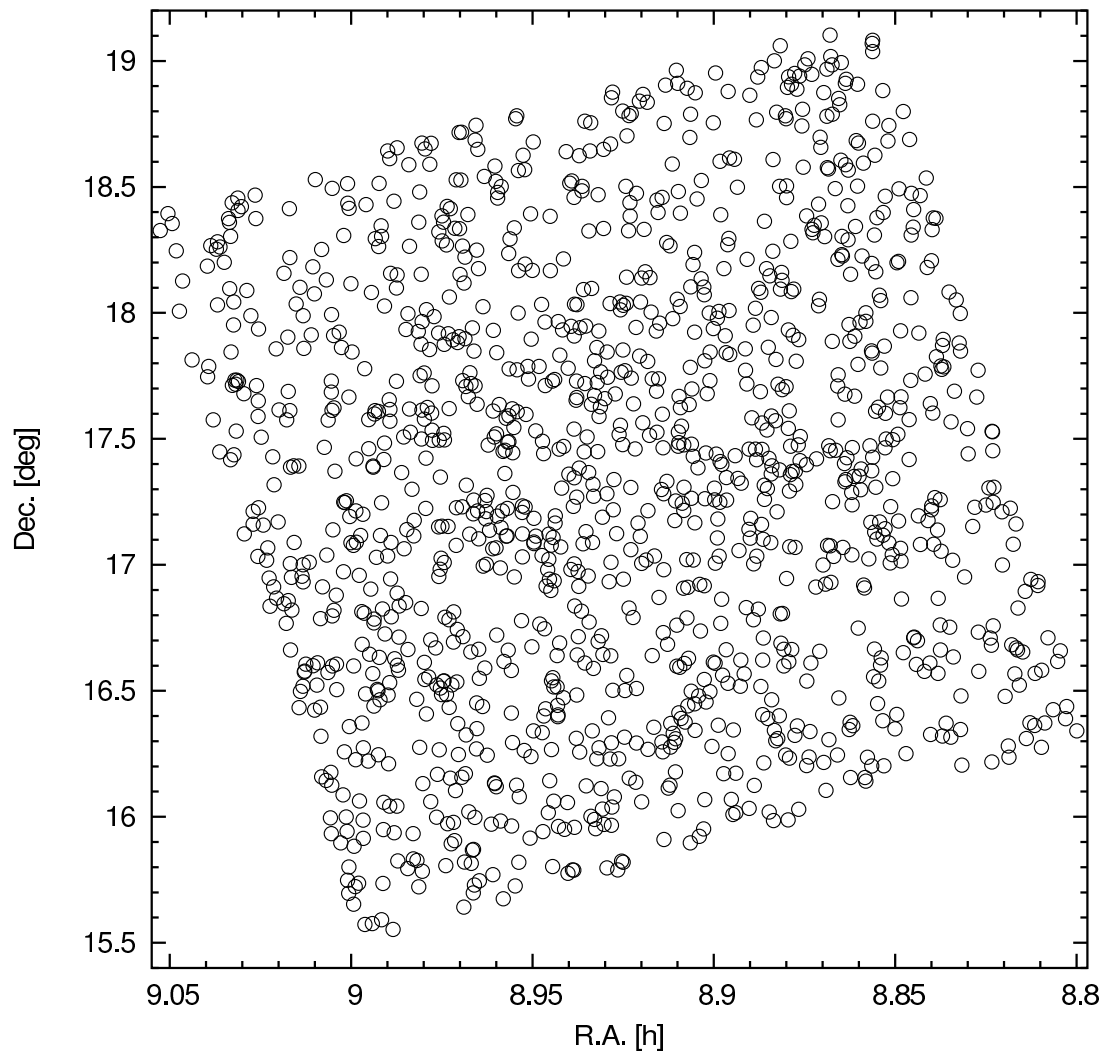


Figure 4.6: Discovery observations of 1313 objects down to $V = 22^m$ in the direction of opposition.

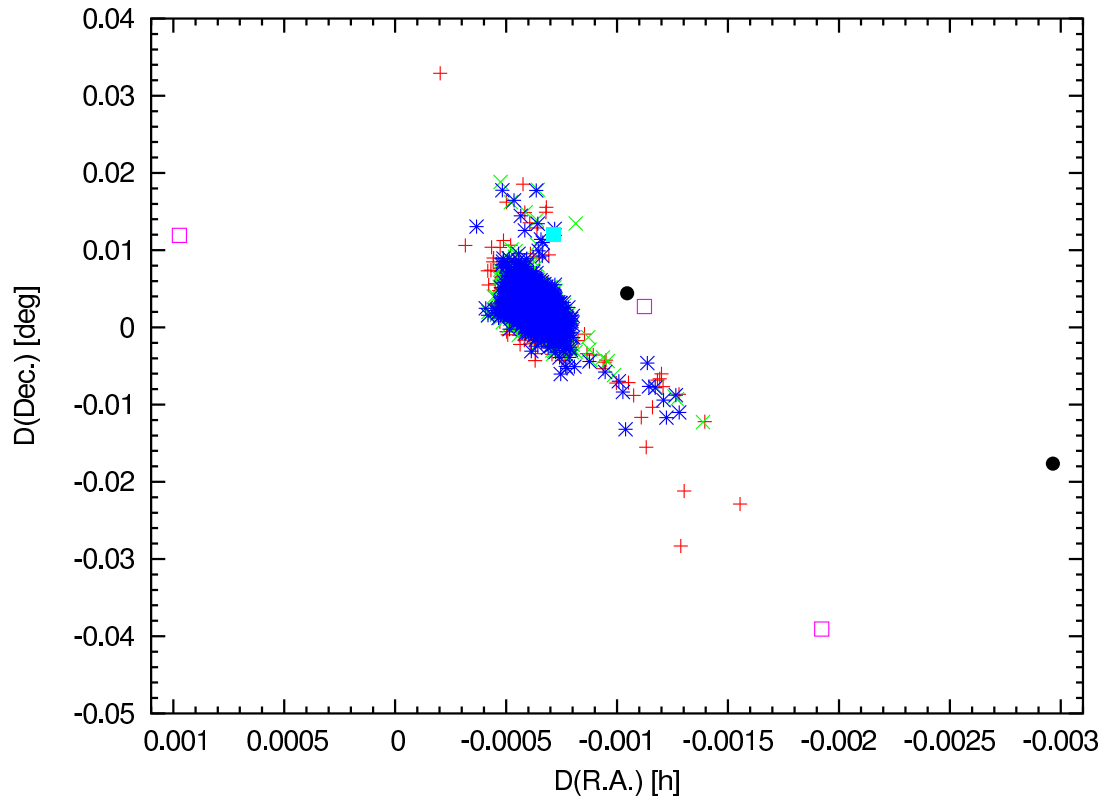


Figure 4.7: Position differences during one hour on the discovery night. Different symbols are used for six separate cases; MBOs at limiting magnitudes 18^m (red plus), 20^m (green cross), 22^m (blue star), and NEOs at limiting magnitudes 18^m (unfilled, magenta square), 20^m (filled, light blue square), 22^m (filled, black circle). Note the small number of NEOs compared to the number of MBOs. The observations are simulated in the direction of opposition. Two NEOs at limiting magnitude 18^m are not displayed because their position differences ($0.007^h, 0.054^\circ$ and $0.005^h, 0.191^\circ$) are outside of the displayed area.

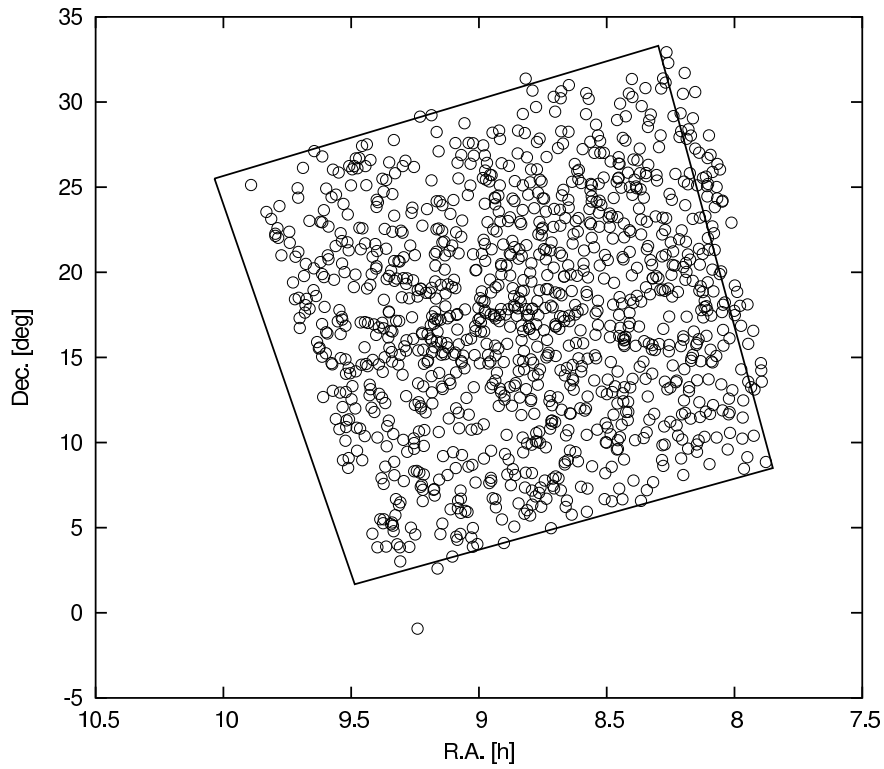


Figure 4.8: The expansion of MBOs during one week. The square marks the observational area ($25^\circ \times 25^\circ$) at discovery. The limiting magnitude is 18^m .

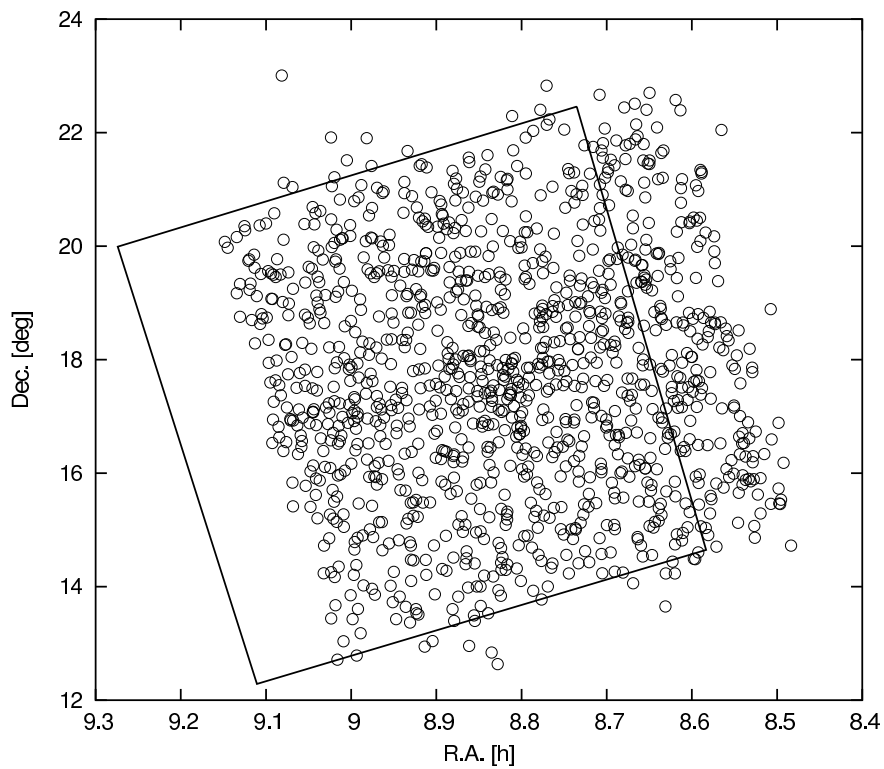


Figure 4.9: The expansion of MBOs during one week. The square marks the observational area ($8^\circ \times 8^\circ$) at discovery. The limiting magnitude is 20^m .

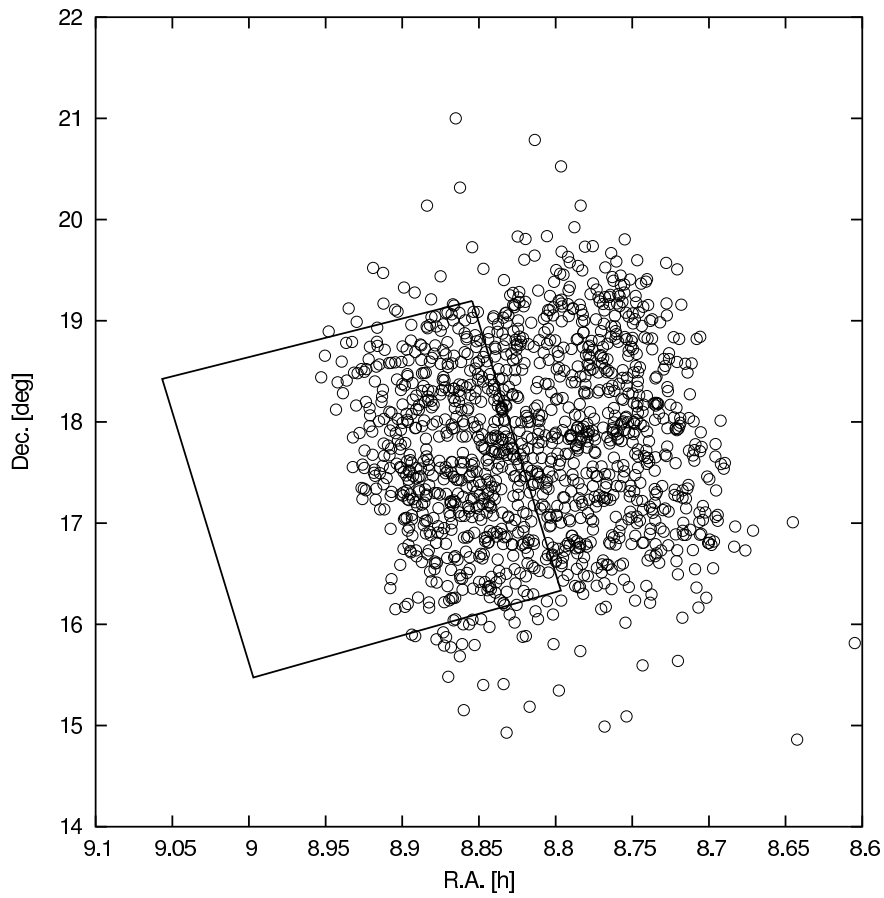


Figure 4.10: The expansion of MBOs during one week. The square marks the observational area ($3^\circ \times 3^\circ$) at discovery. The limiting magnitude is 22^m .

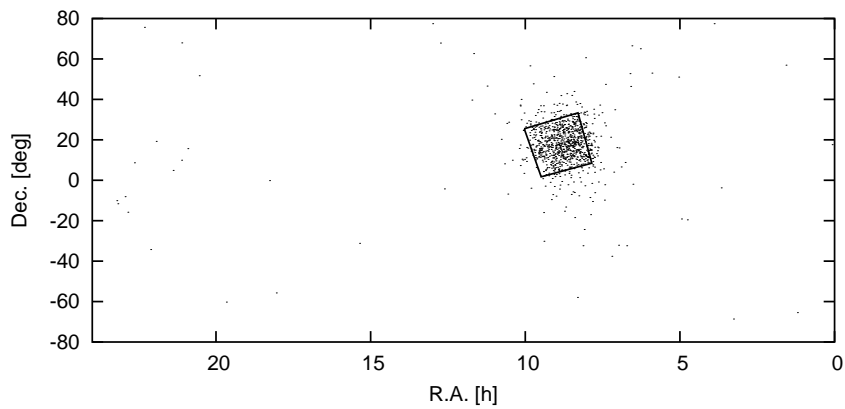


Figure 4.11: The expansion of NEOs during one week. The square marks the observational area ($25^\circ \times 25^\circ$) at discovery. The limiting magnitude is 18^m . Note that the number of objects is not realistic.

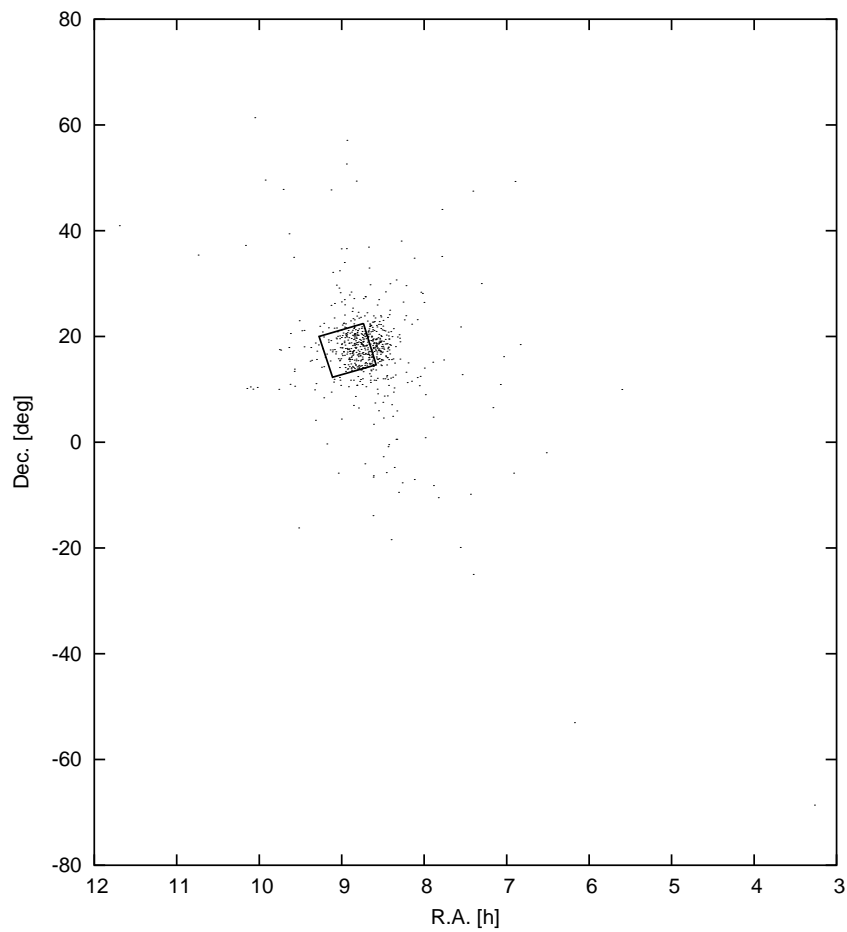


Figure 4.12: The expansion of NEOs during one week. The square marks the observational area ($8^\circ \times 8^\circ$) at discovery. The limiting magnitude is 20^m . Note that the number of objects is not realistic.

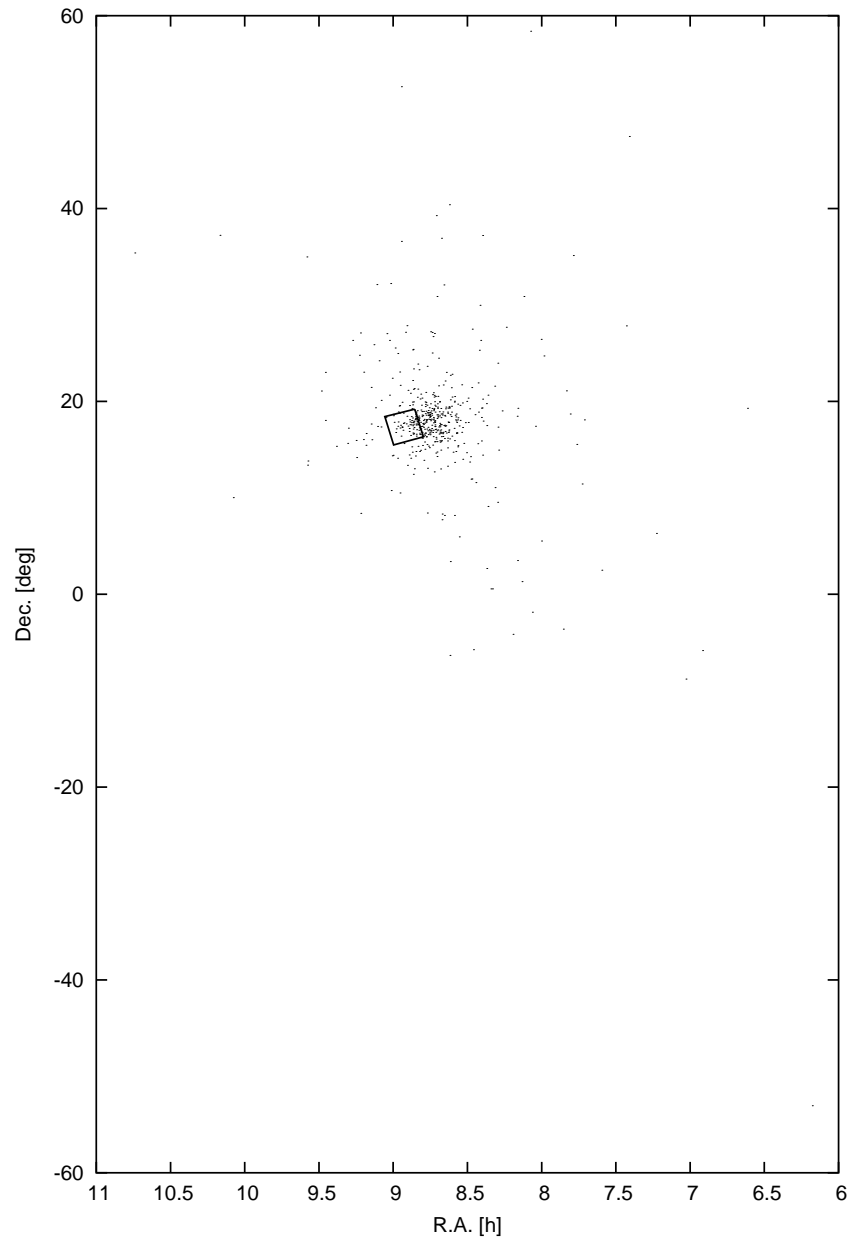


Figure 4.13: The expansion of NEOs during one week. The square marks the observational area ($3^\circ \times 3^\circ$) at discovery. The limiting magnitude is 22^m . Note that the number of objects is not realistic.

Chapter 5

Results and discussion

5.1 Adjusting the parameters

As the building blocks of the identification algorithm are rigorous, the only difficult thing is to find suitable parameters for the procedures. The values of these parameters should simultaneously make the algorithm both accurate and fast.

Of the essential parameters needed by statistical ranging, only the number of orbits and trials were altered. The following values were used for the statistical parameters: the observational noise $\sigma = 0.5''$, the maximum sky-plane residuals were $\pm 3\sigma$ in both R.A. and Dec., and the threshold of $\chi^2 = 30$.

5.1.1 First step — Symmetric ephemeris address comparison

As already mentioned in Section 3.2, the essential duties of the first step is to find all correct identifications and filter out the bulk of the impossible candidates. The key parameters are the number of orbits combined with the bin sizes. Since the ephemeris distributions are discrete, these parameters are linked to each other. So if more orbits are used, the ephemeris distribution will be more dense thus allowing for smaller bin sizes that lead to more efficient filtering and vice versa. Figure 5.1 shows an ephemeris distribution and Figure 5.2 shows the corresponding positional differences after 12 hours for the same object, a very fast moving NEO, but using two different sets of observations. The distributions are spread almost around the sky, yet they overlap at one point. A small part of this area is shown in Figures 5.3 and 5.4.

The bin sizes, or tolerances, and the number of orbits must be adjusted to find correct identifications. This means, that at least one pair of ephemeris arrays, derived from different observational sets, need to fall in the same bin, i.e., acquire the same address. An ephemeris array contains the positions, R.A., and Dec.,

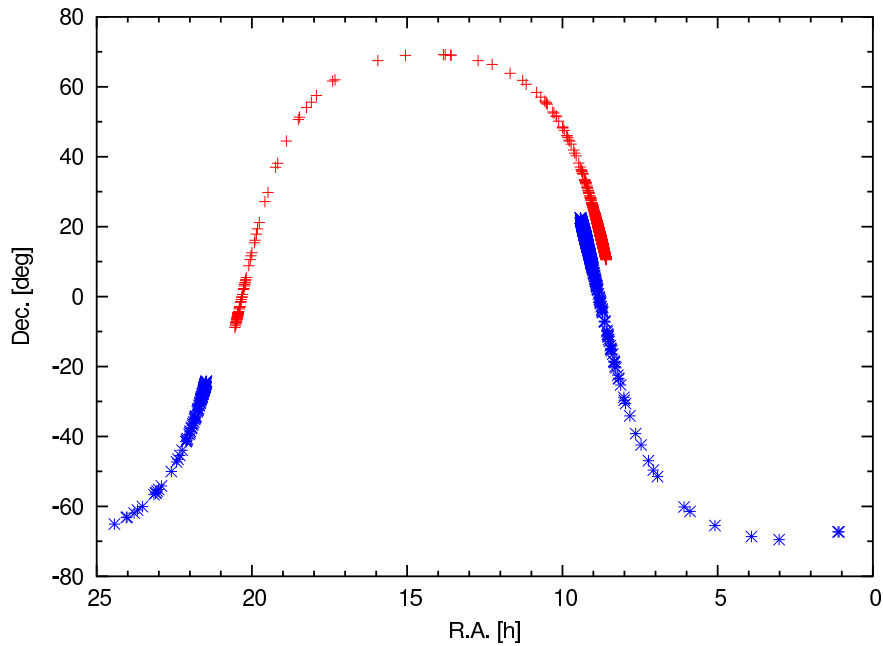


Figure 5.1: An example of two ephemeris distributions for the same fast moving NEO, using different sets of observations with an interval of four days. The time between the two observations in each set is one hour. A $+0.5^{\text{h}}$ shift in R.A., has been added to the blue curve to make the overlap, centered at approximately 8.75^{h} and 17° , visible.

and also the positional differences after 12 hours, $\Delta\text{R.A.}$, and ΔDec .

If the observational sets contain only a few observations, it is possible to find suitable parameters. As long as the ephemeris distributions are spread all around the sky, it is probable that the distance between at least one of the ephemeris arrays is small enough to allow for correct identifications. However, if the sets contain observations of a fast moving target, like in the case above, it might turn out that by adding a few more observations the task becomes more difficult. The ephemeris distribution computed for an object with a long observational arc or several observations, can appear as a point compared to the second distribution, as seen in Figures 5.5 and 5.6.

When adjusting the parameters, the main criterion is that all correct identifications has to be found, even if it means making the algorithm slower. The worst case scenario, which was used for testing the parameters, was assumed to be the *observational scheme 0-7* at limiting magnitude 22^{m} . The parameters allowing for the derivation of all correct identifications are:

Orbits	Tolerances			
	R.A.	Dec.	$\Delta\text{R.A.}$	$\Delta\text{Dec.}$
2000	$10''$	$10''$	$12''$	$12''$

Although the above parameters allow the correct identifications in the worst case scenario, it turns out that the assumed worst case scenario was not the

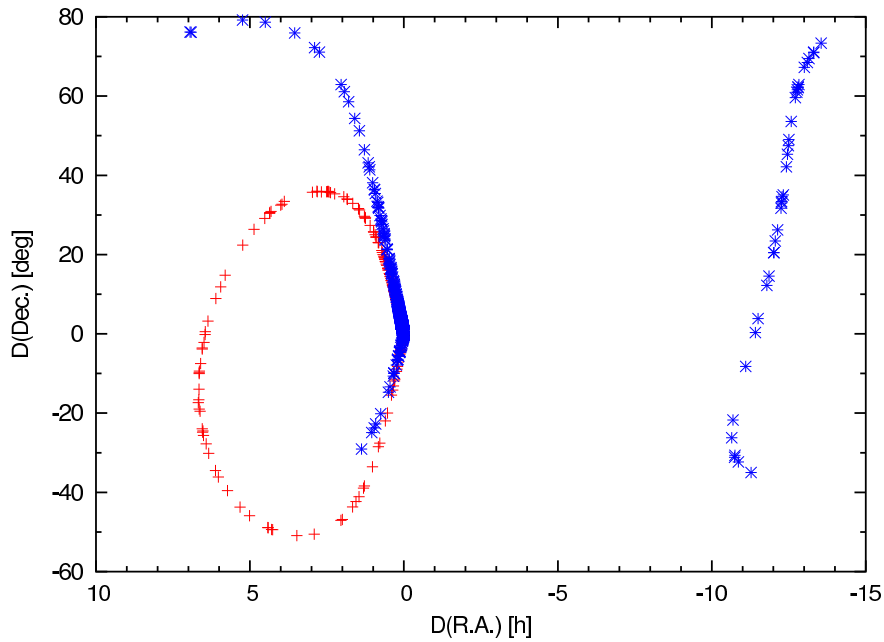


Figure 5.2: Same as Figure 5.1, but showing coordinate differences after a time interval of 12 hours. No shift has been added.

most difficult case. By using the above parameters, the algorithm was unable to detect all correct identifications in the $V = 18^m$ case. Roughly 99% of the correct identifications were found, but some of the most interesting and important NEOs were missed. The reason is shown in Figures 5.5 and 5.6. The only difference between the two generated sets of observations, besides limiting magnitude, is the window size. Obviously the larger window size at $V = 18^m$ allows for the detection of fast moving NEOs thus making this observational set the actual worst case scenario. Therefore, the parameters had to be evaluated once more using a different set of observations. The final parameters are:

Orbits	Tolerances			
	R.A.	Dec.	Δ R.A.	Δ Dec.
2000	40''	40''	48''	48''
	($\sim 0.00074^h$)	($\sim 0.011^\circ$)	($\sim 0.00089^h$)	($\sim 0.013^\circ$)

Assuming that the smaller ephemeris distribution, resulting from a longer observational arc, is always surrounded by the larger ephemeris distribution, it should be possible to find suitable parameters more systematically by computing the maximum distance between neighbouring ephemeris points for large ephemeris distributions derived from many different observations. The outcome of such a computation is naturally dependent on many different parameters like the number of orbits, time span from the observational epoch to the epoch of

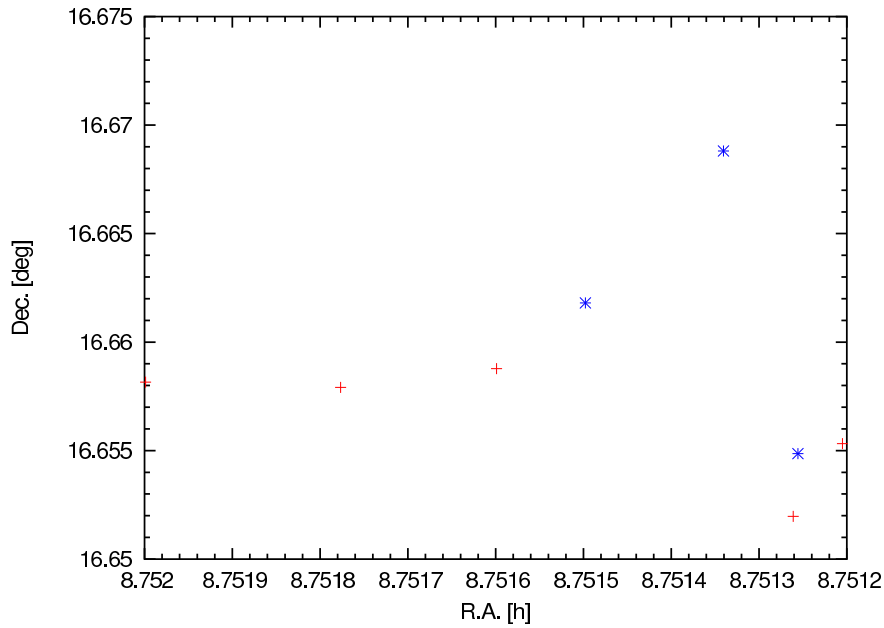


Figure 5.3: A close-up of Figure 5.1 showing two ephemeris distributions for the same object, but using different sets of observations with an interval of four days. The time between the two observations in each set is one hour. The tolerances in R.A., and Dec., have to be large enough to accept this identification, i.e., the angular distance between at least one pair of ephemeris, corresponding to objects in different sets, has to be smaller than the tolerance.

ephemeris, and the set of observations.

5.1.2 Second step — Single-orbit filtering

One specific feature of statistical ranging is its diminishing efficiency, i.e., the ratio of found orbits and trial orbits with increasing orbital and observational arcs. This occurs when the bounds of the topocentric ranges are kept constant. Consider the situation in a six-dimensional orbital element space $\mathbb{E} \in \mathbb{R}^6$. When the amount of data is barely enough to allow for a meaningful inversion, very different orbits fit the observations to predefined accuracy. In other words, the volume mapped by the orbit distribution in \mathbb{E} is substantial. As more observations are added, the difference between orbits fitting to the observations reduces, and the volume mapped in \mathbb{E} decreases. Now, assuming that the topocentric bounds, i.e., the total volume of \mathbb{E} , are constant, it becomes evident that random orbital elements are accepted less frequently with an increasing number of observations. In other words, the Monte Carlo algorithm requires more trials before finding an acceptable solution.

The key parameters for the single-orbit filtering are the number of trials combined with the bounds of the topocentric ranges. As there are many possible

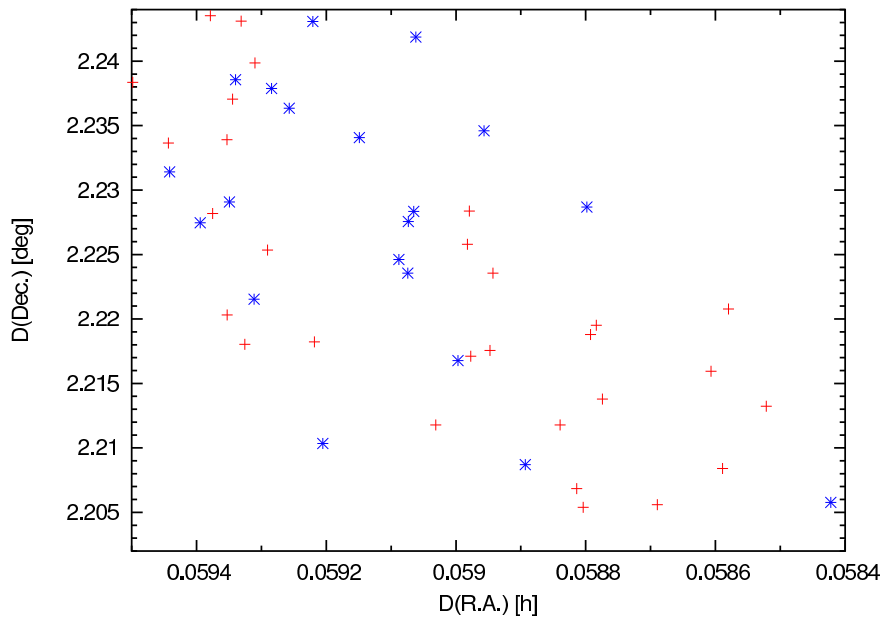


Figure 5.4: Same as Figure 5.3 but a close-up of Figure 5.2, showing position differences after a time interval of 12 hours. The tolerances in $\Delta R.A.$, and $\Delta Dec.$, have to be large enough to accept this identification, i.e., the distance between at least one pair of position differences corresponding to objects in different sets has to be smaller than the tolerance.

identifications that will fail this step, the number of trials must be kept as small as possible to achieve an efficient routine. Understanding the coupling between the amount of trials and the bounds of the topocentric ranges, it is clear that some kind of test is needed to find suitable parameters. Assuming that the bounds of the topocentric ranges are computed similarly for all cases, the only thing that needs to be determined is how many trials are required to find a single solution.

The idea is to generate a number of objects observable in a given window for a period of time. Observational sets are generated by selecting two observations of an object from one night and combining them with two from another night. Then, by generating one orbit from these observations, a number of trials is found. A histogram describing the numbers of trials is finally produced by repeating this method for every object.

Four sets of observations have been generated, two of NEOs and two of MBOs. The time interval between two observational sets is seven days, i.e., the same as the worst case to be examined. As the number of trials increases for longer observational arcs, the windows to be chosen must be of the same size as the largest windows used in the study. The larger the windows become, the longer the observational arcs will be, and thus the process of statistical ranging becomes more difficult. The size of the observational window was the same for both NEOs and MBOs; $25^\circ \times 25^\circ$. It is assumed that approximately 1,000 objects in each

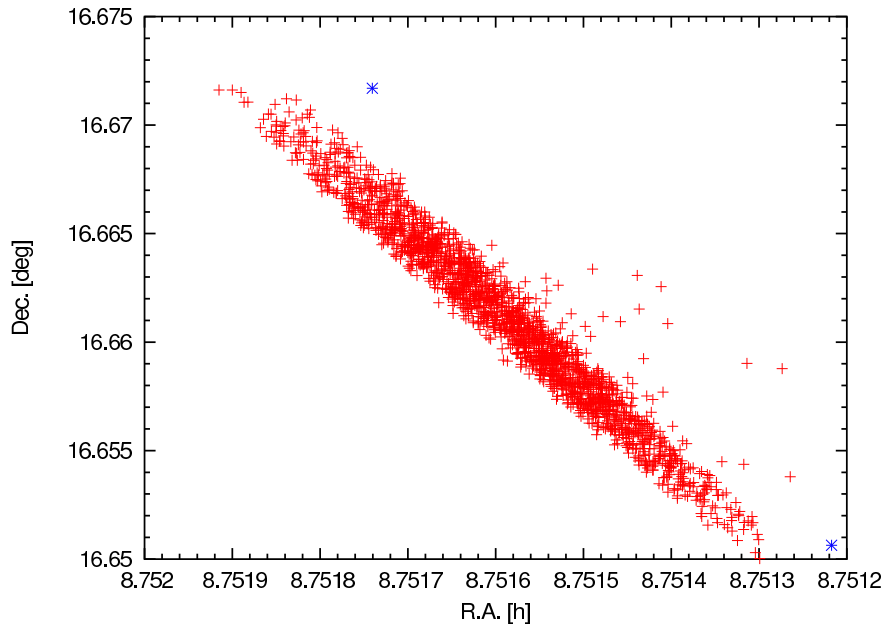


Figure 5.5: Adding two more observations (see Figures 5.1 and 5.3) from the second night dramatically shrinks the first ephemeris distribution. A correct identification can be missed, if a tight ephemeris distribution is located in a less dense region of the second ephemeris distribution.

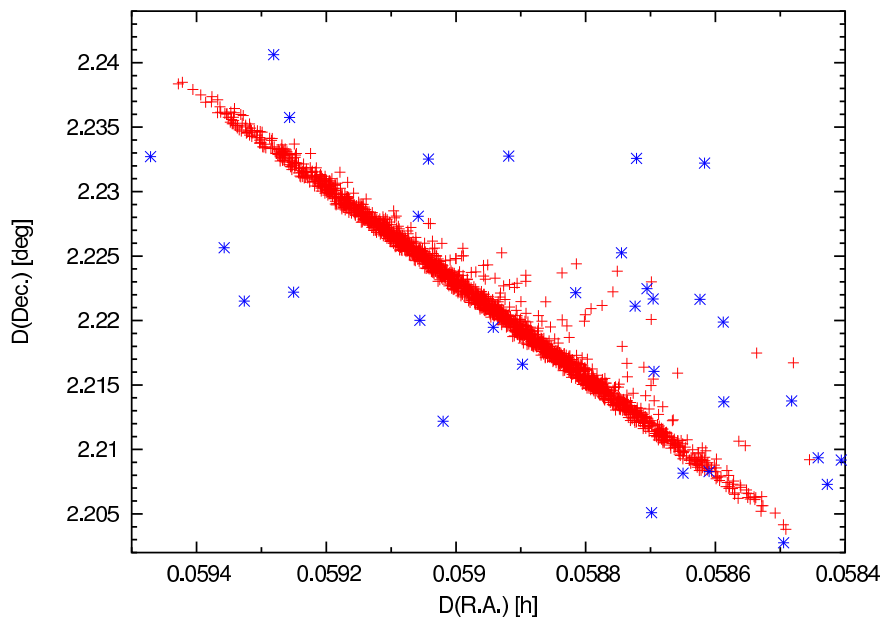


Figure 5.6: Same as Figure 5.5, but showing the positional differences (see Figures 5.2 and 5.4).

population gives a realistic view of suitable values. The absolute magnitudes were 25^m and 17^m , respectively. For MBOs, the generated population for $V = 18^m$ was used (993 objects), while the NEO population was generated using a limiting magnitude of 26.5^m (1,036 objects).

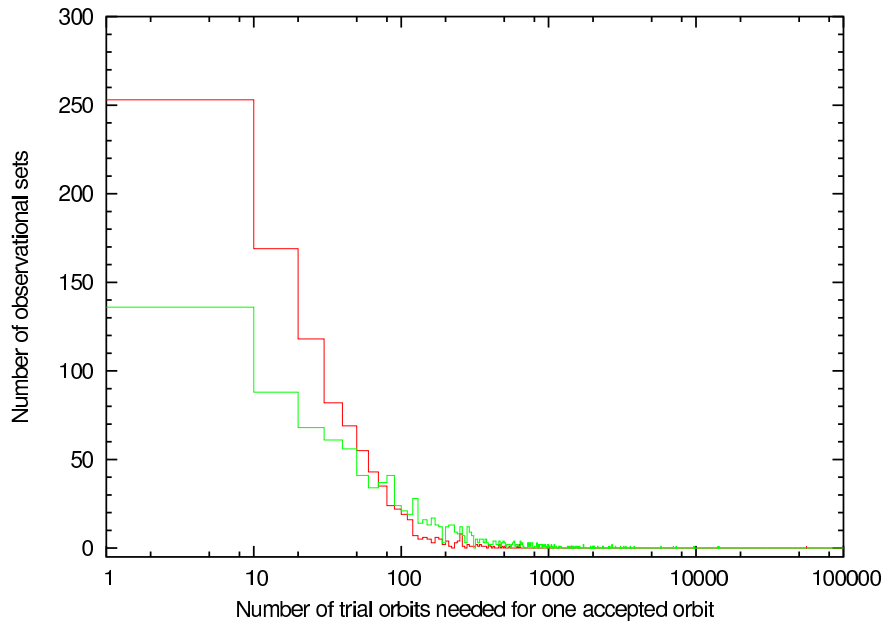


Figure 5.7: Histogram showing the amount of trials needed when generating one orbit using correct identifications and individually chosen topocentric bounds. The observational noise σ and the $\Delta\chi^2$ threshold were the same as during the identification process, i.e., $0.5''$ and 30, respectively. The red curve marks MBOs (993 objects) and the green marks NEOs (1,036). The largest numbers of trials were 56,057 for MBOs, and for one NEO more than 100,000 trials were required and this went over the scale. These two objects are real oddities, because the number of trials usually needed is much smaller. The observational data consisted of two sets of two observations. The time span between sets was seven days, and the sizes of the observational windows were $25^\circ \times 25^\circ$, centered in the direction of opposition at the first epoch.

The largest number of trials was greater than 100,000 for NEOs, and was 56,057 for MBOs. As can be seen in Figure 5.7, the numbers were usually much smaller. As a compromise between accuracy and speed, it was decided that the number of trials in the single-orbit filtering be set at 25,000.

5.1.3 Third step — Orbit distribution filtering

Since the main purpose of this step is to demonstrate the potential for producing an orbit distribution using observations of a possible identification, the only key parameter is the number of orbits. It should, however, be pointed out that in some situations, e.g., in cumulative identification runs, also the number of trials makes a notable difference. The reason, which was already pointed out in Section 5.1.2, is that statistical ranging gets slower as the number of observations, or the observational arc, increases. The number of trials is chosen to be 1,000,000.

When using automated statistical ranging, the number is irrelevant and can

be, e.g., 1 or 10,000. Typically, a set of observations that can be inverted to one orbit can also be inverted to 10,000 orbits using the same input values. The reason behind this is the stepwise use of statistical ranging. The number of orbits is chosen to be one, unless stated otherwise. Note that the automated version is not used when the number of orbits is set to one.

5.2 Identification of single-night asteroids

Objects that are not observed during a second night within a week after discovery, are called single-night asteroids. So far many of these objects have been almost impossible to identify with each other, as there are only a few observations available for each of them. Even if some of these have been identified with previously known objects (Milani et al. 2001), there could be a large amount of objects that have actually been observed multiple times, but the identification has proved either impossible (too few observations), or too difficult (nonlinear effects).

As there are no limitations with the current identification algorithm relating to the number of observations or the nonlinearity of the situation, it can be used for searching identifications among single-night asteroids. Table 5.1 displays the results of identifications among simulated single-night asteroids using different observational schemes and limiting magnitudes. The algorithm finds all correct identifications in the first step, but as is evident, all correct identifications are not found in the second step, and consequently not in the third either. The overall rate of success in finding correct identifications is, however, excellent, as more than 99.7% of them are detected in all cases.

The reason for the lacking detections is that the number of trial orbits in the second step is too small. The observations, that can not be linked despite being correct identifications, are all related to MBOs. For instance, when the number of trials is 100,000, instead of 25,000, all correct identifications are found for scheme 0-7 at limiting magnitude 18^m . The price that has to be paid, however, is a slightly slower routine. Note that for some cases the number of false positive identifications varies between the second and the third step. This how the statistical nature of the algorithm is manifested; in the second step one orbit is found using no more than 25,000 trials, but in the third step 1,000,000 trials are not enough!

Figures 5.8 and 5.9 display the percentage of false positive identifications with respect to correct identifications as function of time span between observational sets. The first figure displays the situation after the first step, i.e., after symmetric ephemeris address comparison, and the second figure displays the final

situation after the third step. As there are only minor differences between the percentages between the second and the third step, the situation after the second step is not displayed separately.

Scheme	Correct	Step 1		Step 2		Step 3	
	total	correct	false	correct	false	correct	false
Limiting magnitude 18 ^m							
0-1	1146	1146	1	1144	0	1144	0
0-2	1122	1122	7	1121	0	1121	0
0-3	1089	1089	14	1089	2	1089	2
0-4	1064	1064	30	1063	4	1063	4
0-5	1040	1040	47	1039	7	1039	7
0-6	1017	1017	77	1016	8	1016	8
0-7	994	994	103	993	11	993	11
Limiting magnitude 20 ^m							
0-1	1167	1167	41	1166	0	1166	0
0-2	1095	1095	114	1095	9	1095	9
0-3	1038	1038	214	1037	27	1037	27
0-4	980	980	351	979	40	979	40
0-5	925	925	519	923	70	923	70
0-6	845	845	679	844	87	844	87
0-7	790	790	871	788	121	788	121
Limiting magnitude 22 ^m							
0-1	1162	1162	321	1160	21	1160	21
0-2	993	993	807	992	94	992	94
0-3	859	859	1482	857	172	857	172
0-4	721	721	2179	721	261	721	261
0-5	611	611	2979	611	367	611	366
0-6	489	489	3468	489	458	489	458
0-7	372	372	3379	372	465	372	463

Table 5.1: Identification results among single-night asteroids. The steps refer to different filters; the first step uses symmetric ephemeris filtering, the second step single-orbit filtering, and the third step orbit distribution filtering. The total numbers of correct identifications for each scheme and magnitude are given in the *Correct total* column, while the numbers of correct identifications found during each step are shown in the *correct* columns, and false positive identifications in the *false* columns.

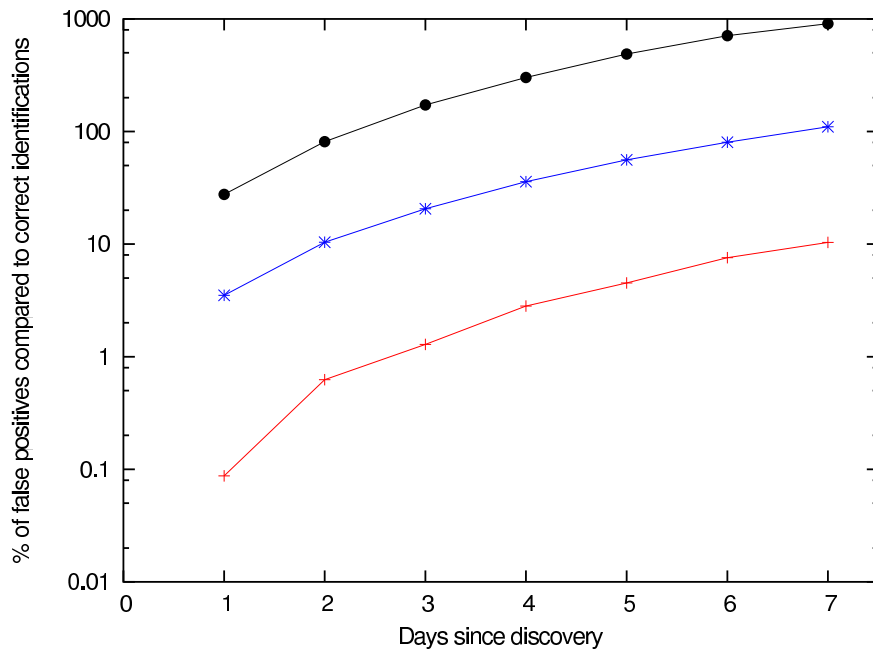


Figure 5.8: The percentage of false positive identifications with respect to correct identifications as function of time span between observational sets after the first step, i.e., after symmetric ephemeris address comparison. Different limiting magnitudes, V , are displayed with different symbols: $V = 18^m$ is displayed with red pluses, $V = 20^m$ with blue stars and $V = 22^m$ with black dots. The points are connected with lines to make the trend more clearly visible.

5.3 Cumulative identification and observational strategies

Cumulative identification is an identification procedure close to current practical problems, compared to identifying single-night asteroids separated by several days. The word cumulative stands for a stepwise identification procedure. If there are observations from several nights, e.g., the first, the second, and the third night, the identification algorithm is initially applied to the first and second nights, and then the possible identifications are processed with the third night's data.

Since there are lots of possibilities¹ on how to choose three or more nights for examination from a total of eight nights, not all of them can be covered. Two well-known strategies were therefore chosen to provide the observational schemes. The first strategy is the one recommended by the Minor Planet Center (MPC), and the second is a generic one used by automatic surveys. It should be noted that the strategy promoted by the MPC is an idealisation of reality. It is often impossible to make real observations precisely as suggested, due to,

¹Exactly $6! 5! 4! 3! 2! 1! = 24, 883, 200$ different possibilities.

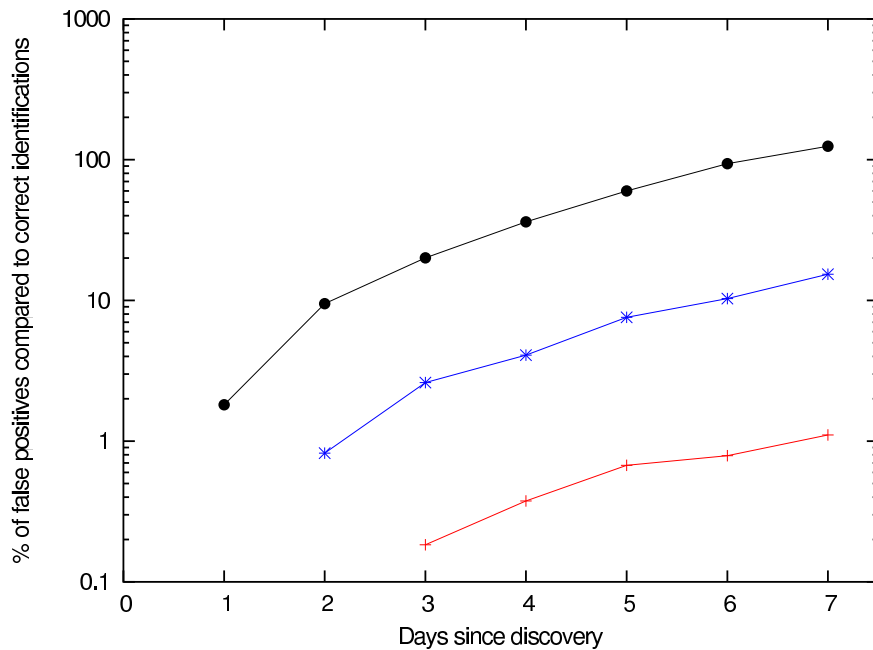


Figure 5.9: The percentage of false positive identifications with respect to correct identifications as function of time span between observational sets after the final step. Different limiting magnitudes, V , are displayed with different symbols: $V = 18^m$ is displayed with red pluses, $V = 20^m$ with blue stars and $V = 22^m$ with black dots. The percentage is zero for schemes 0-1 and 0-2 at $V = 18^m$, and for scheme 0-1 at $V = 20^m$. The points are connected with lines to make the trend more clearly visible.

for instance, weather conditions. The generic automatic survey strategy, on the other hand, gives a good approximation of real observations as it is consistently used in automated surveys.

Unidentified targets from the first part of the process are regarded as lost objects with respect to the forthcoming identification rounds. The simple reason is that asteroids leaving the frame will seldom return to the same frame within a short period of time.

5.3.1 Minor Planet Center

In the *Guide to Minor Body Astrometry*² the MPC suggests that newly discovered objects should be observed more or less according to the scheme 0-1-8-30-... , i.e., two or three observations should be made on the night following the discovery night, a week after discovery, and a month after discovery, and so forth. By promoting an observational strategy, the MPC staff makes an implicit statement that claims they are able to handle observations produced according to the recommendations. Having a rigorous algorithm at hand, it is interesting to assess

²At <http://cfa-www.harvard.edu/iau/info/Astrometry.html>

whether these implicit claims are justified when it all comes around. Recently, the MPC has changed the guidelines to now support a 0-1-8-9-30-31-scheme. This change in guidelines is not significant, so using the old scheme is satisfactory as it will still provide hints on the reliability of identifications produced by such observations. Note that the scheme used here (0-1-7) is even slightly more compressed, with regard to the time span, than the one promoted by the MPC. In other words, the number of false positives acquired would be even greater if doing it according to the MPC guidelines.

Limiting magnitude	Step 1		Step 2		Step 3	
	correct	false	correct	false	correct	false
18	992	20	991	7	982	0
20	789	186	787	78	785	10
22	371	848	371	307	366	22

Table 5.2: Scheme 0-1-7.

It seems that the percentage of false positive identifications with respect to correct identifications would be of the order of $\sim 0\%$, 1% , and 6% , at limiting magnitudes 18^m , 20^m , and 22^m , respectively (Table 5.2).

5.3.2 Automatic surveys

A generic automatic survey observes the same region two to five times during a single night and returns to the same area every three to four days (Stokes et al. (2003) and private communication with T. Grav).

Limiting magnitude	Step 1		Step 2		Step 3	
	correct	false	correct	false	correct	false
18	994	10	993	7	957	0
20	789	120	787	83	778	1
22	372	396	372	265	367	10

Table 5.3: Scheme 0-3-7.

The percentage of false positive identifications, with respect to correct identifications, are of the order of $\sim 0\%$, 0.1% , and 3% , for limiting magnitudes 18^m , 20^m , and 22^m , respectively (Table 5.3).

5.4 Interpretation

Firstly, of all it should be stressed that a single set of simulated observations obtained with a statistical tool is not statistically significant. To ensure valid results, the entire examination, i.e., from generating observations to finding identifications, should be repeated several times. Finally, the mean value of all individual results should be calculated and presented. In this study, the results are based on one of the individual results, leading to only order-of-magnitude predictions.

As pointed out in Section 3.4, all of the possible identifications fit the observational data to the assumed observational error, which in this study is well known. If there are suspicions on whether an identification is correct, it should be examined with further observations. It is also possible, however, to estimate the probability of a correct identification by computing a value describing the goodness of the fit. As seen in Figure 5.10, correct and erroneous identifications seem to form two groups of χ^2 values. Unfortunately, the overlap between these groups makes it impossible to use the χ^2 when deciding if suspicious identifications are false positives, or not. Large χ^2 values do seem to indicate false positive identifications though.

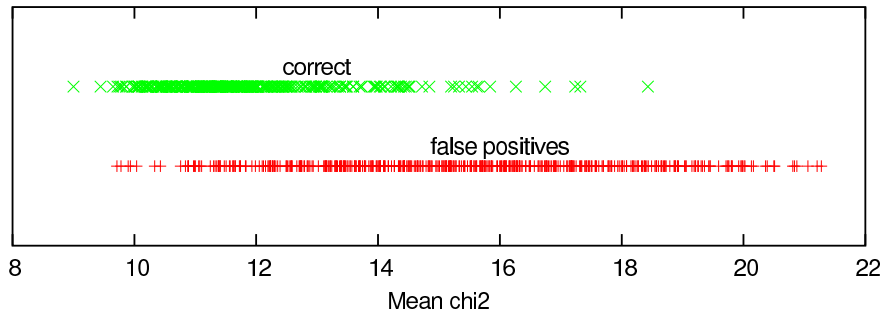


Figure 5.10: Mean χ^2 values for the worst scenario case. The mean of χ^2 is computed for each identification by taking into account the χ^2 value of every single orbit in the orbit distribution (here 100 orbits). Limiting magnitude is 22, and the time span between the observational sets is one week. Correct identifications (green x) usually fit better to the observations, which generally leads to smaller χ^2 values compared to erroneous identifications (red +). There does not, however, exist any strict boundary value between these groups.

The percentage of false positive identifications versus correct identifications used in the examinations should be interpreted as depicting the whole sample instead of individual identifications. Individual error margins can be obtained by evaluating χ^2 values that describe the probability of the identification.

The outcome of Section 5.3 suggests that the observation strategy promoted by the MPC might not be the ultimate solution for identification purposes. Even

if the generic observation strategy used by automatic surveys appear to be optimal from the identification point of view, there may be even better solutions still to be discovered. Using the exact method described here, it should be possible to find the best strategy. Unfortunately, it is beyond the scope of this thesis and must therefore be put aside for the time being.

All simulations were run on AMD Athlon 1.4GHz single-processor Linux PCs. Comparison of computational loads is difficult, because the used computers are shared ones. Generally, there appears to be a huge difference in the required cpu-time between single-night identification runs and cumulative identification runs. It is, however, not a surprise, as the algorithm was optimised for identification among single-night asteroids.

For single-night identification runs, the cpu-time needed for the first step varied from one hour to a few hours. The second step was usually the fastest, and it was performed in time scales from seconds up to a few hours. Statistical ranging slows down when the observational arc or the number of observations increases. When the observational arc got longer in the third step, the cpu-time usage increased substantially. For instance, when going from scheme 0-3 to scheme 0-7 it roughly doubled.

In case of cumulative identification runs, the first and the second step was approximately as fast as among single-night identification runs. The computations took substantially longer during the final step due to the increased amount of observations. Typically, the step took some 12 to 36 hours. The cpu-time consumption at the final step could probably be decreased by optimising the parameters of the entire identification procedure, and perhaps an additional step could be inserted between the second and the third step. The task of the new filtering step would be to establish a single solution, like that of the second step, but using all observations available. This would allow the last step to deal with observational sets for which the inversion practically never fails.

Chapter 6

Conclusions

As far as it is known, the algorithm presented in this paper is the first published and probably the only tool suitable for solving the identification problem for the shortest observational arcs and the smallest possible numbers of observations. This algorithm is capable of producing identifications among single-night asteroids, which, up to now, has been impossible. The algorithm has revealed its power during the most systematic and accurate tests published to date, and with minor modifications, it is poised for real identification problems that are beyond the of scope of the thesis.

Despite accusations that the statistical methods were too slow for routine use, the algorithm was able to demonstrate outstanding speed and reliability. Applying a modest PC over a few hours, for the rigorous filtering of more than one million possible identifications, should satisfy all current needs. The quantity of data obtained with the new automatic telescopes will probably be too large to handle with individual PCs, thus forcing supercomputers or clusters of PCs to take the responsibility. Currently, identifications are obviously done, at least in part, manually at the MPC. This will be impossible in the future and as everyone involved is aware, this impossibility is already apparent. The solution is to use exact methods that allow for automatic processing throughout the entire procedure. Even if these methods, e.g., statistical ranging, are slightly slower than the methods used today, the ability to solve problems without human intervention will allow faster execution in the long run.

Due to its rigorous treatment of the problem, the algorithm presented here can be used to evaluate observation strategies from the identification point of view. According to the tests described in Section 5.3, the strategy promoted by the MPC is not as good as the generic one used by automated surveys. However, due to small differences and small observational sets, it is too early to say whether this is a real effect or just a coincidence.

The terminology by Milani (1999), which was presented in Section 1.2, can be

revised, as the use of statistical ranging and other forthcoming statistical methods allows us to solve the inversion problem for any set of observations. Thus, the key question is not whether the inversion *can be done*, but *which method should be used* in the inversion and identification. The different methods for identification could be the following three:

- Position and motion at mid-date.
- Orbital elements (5–6) at mid-date.
- Orbital elements (Milani et al. 2000).

A question concerning the future of the identification problem arose during the process based on the observation that as the limiting magnitude increases, the positional difference between observations get smaller and smaller. Assuming that the observational accuracy does not improve in the future, it is perceivable that the observational errors could become larger than the mean positional difference between objects. Will absolute identification be possible after this point? If not, what kind of implications will it have on other predictions such as impact probabilities? Will the critical depth ever be reached due to Gegendchein, ever improving accuracy, or finite number of objects?

References

- Bowell, E., Chernykh, N. S. & Marsden, B. G. (1989), *Asteroids II*, University of Arizona Press, chapter ‘Discovery and follow up of asteroids’, pp. 21–38.
- Bowell, E. & Muinonen, K. (1994), *Hazards due to Comets and Asteroids*, University of Arizona Press, chapter ‘Earth-crossing asteroids and comets: ground-based search strategies’, pp. 149–197.
- Bowell, E., Muinonen, K. & Wasserman, L. H. (1994), A Public-Domain Asteroid Orbit Data Base, *in* A. Milani, M. di Martino & A. Cellino, eds, ‘Asteroids, Comets, Meteors 1993’, Proceedings of IAU Symposium 160, Co-published by IAU and Kluwer Academic Publishers, Dordrecht, p. 477.
- Bowell, E., Virtanen, J., Muinonen, K. & Boattini, A. (2003), *Asteroids III*, University of Arizona Press, chapter ‘Asteroid Orbit Computation’, pp. 27–43.
- Chesley, S. (2003), ‘Monte Carlo uncertainty estimation’, Minor Planet Mailing List. <http://groups.yahoo.com/group/mpml/message/9488>.
- Danby, J. M. A. (1992), *Fundamentals of Celestial Mechanics*, 2 edn, Willman-Bell, Inc., Richmond, Virginia, U.S.A.
- Decyk, V. K., Norton, C. D. & Szymanski, B. K. (1997), Introduction to Object-Oriented Concepts using Fortran90, <http://exodus.physics.ucla.edu/Fortran95/>.
- Haataja, J., Rahola, J. & Ruokolainen, J. (2001), *Fortran 90/95*, 3 edn, CSC - Tieteellinen laskenta Oy, Espoo, Finland.
- Jedicke, R., Larsen, J. & Spahr, T. (2003), *Asteroids III*, University of Arizona Press, chapter ‘Observational Selection Effects in Asteroid Surveys and Estimates of Asteroid Population Sizes’, pp. 71–87.
- Jewitt, D. C. & Luu, J. X. (1993), ‘Discovery of the candidate Kuiper belt object 1992 QB1’, *Nature* **362**(6422), 730–732.

- Karttunen, H. (1994), *Datan käsittely*, CSC - Tieteellinen laskenta Oy, Espoo, Finland.
- Karttunen, H. (2001), *Johdatus taivaanmekaniikkaan*, Tähtitieteellinen yhdistys Ursa ry, Helsinki, Finland.
- Karttunen, H., Donner, K. J., Kröger, P., Oja, H. & Poutanen, M. (2003), *Tähtitieteen perusteet*, 4 edn, Tähtitieteellinen yhdistys Ursa ry, Helsinki, Finland.
- Metcalf, M. & Reid, J. (2002), *Fortran 90/95 explained*, 2 edn, Oxford University Press.
- Milani, A. (1999), ‘The Asteroid Identification Problem I. Recovery of Lost Asteroids’, *Icarus* **137**(2), 269–292.
- Milani, A., La Spina, A., Sansaturio, M. E. & Chesley, S. R. (2000), ‘The Asteroid Identification Problem III. Proposing Identifications’, *Icarus* **144**(1), 39–53.
- Milani, A., Sansaturio, M. E. & Chesley, S. R. (2001), ‘The Asteroid Identification Problem IV. Attributions’, *Icarus* **151**(2), 150–159.
- Milani, A. & Valsecchi, G. B. (1999), ‘The Asteroid Identification Problem II. Target Plane Confidence Boundaries’, *Icarus* **140**(2), 408–423.
- Morrison, D., Harris, A. W., Sommer, G., Chapman, C. R. & Carusi, A. (2003), *Asteroids III*, University of Arizona Press, chapter ‘Dealing with the Impact Hazard’, pp. 739–754.
- Muironen, K. (1998), ‘Discovery and follow-up simulations for small Earth-crossing asteroids’, *Planetary and Space Science* **46**, 291–297.
- Muironen, K. (1999), Asteroid and comet encounters with the earth, in B. A. Steves & A. E. Roy, eds, ‘The Dynamics of Small Bodies in the Solar System’, Kluwer Academic, Dordrecht Norwell, MA, pp. 127–158.
- Muironen, K. (2003), ‘ASurv’, Observatory, University of Helsinki. Asteroid survey simulation software.
- Muironen, K. & Bowell, E. (1993), ‘Asteroid orbit determination using Bayesian probabilities’, *Icarus* **104**(2), 255–279.
- Muironen, K., Bowell, E. & Wasserman, L. H. (1994), ‘Orbital uncertainties of single-apparition asteroids’, *Planet. Space Sci.* **42**, 307–313.
- Muironen, K., Virtanen, J. & Bowell, E. (2001), ‘Collision probability for earth-crossing asteroids using orbital ranging’, *Celestial Mechanics and Dynamical Astronomy* **81**(1–2), 93–101.

- Nesvorný, D., Bottke Jr., W. F., Dones, L. & Levison, H. F. (2002), ‘The recent breakup of an asteroid in the main-belt-region’, *Nature* **417**(6890), 720–722.
- Ostro, S. J. (1989), *Asteroids II*, University of Arizona Press, chapter ‘Radar observations of asteroids’, pp. 192–212.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. (1992), *Numerical Recipes in Fortran — The Art of Scientific Computing*, 2 edn, Cambridge University Press.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. (1999), *Numerical Recipes in Fortran 90 — The Art of Parallel Scientific Computing*, 2 edn, Cambridge University Press.
- Roy, A. E. (1988), *Orbital Motion*, 3 edn, Institute of Physics Publishing, Bristol.
- Standish, E. M. (1998), ‘JPL Planetary and Lunar Ephemerides’, CD-ROM.
- Stokes, G. H., Evans, J. B. & Larson, S. M. (2003), *Asteroids III*, University of Arizona Press, chapter ‘Near-Earth Asteroid Search Programs’, pp. 45–54.
- Tedesco, E. F., Muinonen, K. & Price, S. D. (2000), ‘Space-based infrared near-Earth asteroid survey simulation’, *Planetary and Space Science* **48**, 801–816.
- Virtanen, J., Muinonen, K. & Bowell, E. (2001), ‘Statistical Ranging of Asteroid Orbits’, *Icarus* **154**(2), 412–431.
- Virtanen, J., Tancredi, G., Muinonen, K. & Bowell, E. (2003), ‘Orbit computation for transneptunian objects’, *Icarus* **161**(2), 419–430.

Appendix A

Array-integer transformation algorithms

A.1 Transforming an array to an integer

The purpose of the routine is to transform an array of real numbers to an integer. The idea is to transform each element to an integer as in a basic binning algorithm. Instead of returning the separate integers describing the location of the array in a multidimensional grid, they are presented as a single, often great, integer.

The implementation of the method in Fortran 95 is given below. Note, that the following routine is for the generic case. It allows one to input one-dimensional arrays of any length, and to choose any elements in the array for encoding. The chosen elements are given as a logical array, where `.true.` indicates that the corresponding element should be included in the encoded array.

```
SUBROUTINE arrayToInteger(array,elements,bin_size,bins,bounds,intgr)
```

```
    IMPLICIT NONE
```

```
    REAL, DIMENSION(:), INTENT(in)           :: array      ! input array
    LOGICAL, DIMENSION(:), INTENT(in)        :: elements    ! used elements
    REAL, DIMENSION(:), INTENT(in)           :: bin_size    ! bin sizes
    INTEGER, DIMENSION(:), INTENT(in)        :: bins        ! number of bins
    REAL, DIMENSION(:, :), INTENT(in)        :: bounds      ! variable bounds
    INTEGER(8), INTENT(out)                   :: intgr       ! output integer
    INTEGER(8), DIMENSION(:), ALLOCATABLE    :: kp
    INTEGER, DIMENSION(:), ALLOCATABLE       :: help, bins_tmp
    INTEGER(8)                                :: bins_coeff
    INTEGER                                    :: ncolumn, i, j
```

```

! Make an index (help) of the elements that are used:
ncolumn = COUNT(elements)
ALLOCATE(kp(ncolumn),help(ncolumn),bins_tmp(ncolumn))
j = 0
DO i=1,SIZE(elements)
  IF (elements(i)) THEN
    j = j + 1
    help(j) = i
    bins_tmp(j) = bins(i)
  END IF
END DO

! Compute coordinates of the bin:
DO i=1,SIZE(kp)
  ! Compute a value describing a single elements position:
  kp(i) = &
    INT((array(help(i)) - bounds(help(i),1)) / &
    bin_size(help(i)),4) + 1_8
END DO

! If one or more of the elements are out of bounds, skip the
! array:
IF (ANY(kp>bins_tmp) .OR. ANY(kp<0)) THEN
  intgr = 0_8
  RETURN
END IF

! Transform bin coordinates to a single integer value
! (size(kp) == size(help)):
intgr = 0_8
DO i=1,SIZE(kp)
  bins_coeff = 1_8
  DO j=2,i
    bins_coeff = bins_coeff*INT(bins_tmp(j-1),8)
  END DO
  intgr = intgr + INT((kp(i) - 1),8)*bins_coeff
END DO
intgr = intgr + 1_8

DEALLOCATE(kp, help, bins_tmp)

```

```
END SUBROUTINE arrayToInteger
```

A.2 Transforming an integer to an array

The routine does the same as the one presented in Section A.1, but the other way round. In other words, an integer is transformed to an array. Note that if an array is first transformed to an integer, and then back to an array, information will usually be lost. The same happens with all binning algorithms, and the magnitude of the loss is proportional to the bin size: using smaller bin sizes leads to smaller losses in accuracy.

The following implementation of the algorithm in Fortran 95 is generic, similarly as the implementation in Section A.1.

```
SUBROUTINE integerToArray(intgr,elements,bin_size,bins,bounds,array)
```

```

IMPLICIT NONE
INTEGER(8), INTENT(in)           :: intgr    ! input integer
LOGICAL, DIMENSION(:), INTENT(in) :: elements ! used elements
REAL, DIMENSION(:), INTENT(in)   :: bin_size ! bin sizes
INTEGER, DIMENSION(:), INTENT(in) :: bins     ! number of bins
REAL, DIMENSION(:,,:), INTENT(in) :: bounds  ! variable bounds
REAL, DIMENSION(:), INTENT(out)  :: array    ! output array
INTEGER(8)                       :: intgr_coeff, bins_coeff
INTEGER, DIMENSION(:), ALLOCATABLE :: kp, help, bins_tmp
INTEGER                           :: ncolumn, i, j, k

! Make an index (help) of the elements that are used:
ncolumn = COUNT(elements)
ALLOCATE(kp(ncolumn), help(ncolumn), bins_tmp(ncolumn))
j = 0
DO i=1,SIZE(elements)
  IF (elements(i)) THEN
    j = j + 1
    help(j) = i
    bins_tmp(j) = bins(i)
  END IF
END DO

```

```

! Transform integer to bin coordinates
! (size(kp) == size(help)):
bins_coeff = 1_8
DO i=ncolumn, 1, -1
  intgr_coeff = intgr
  DO j=i+1, ncolumn
    bins_coeff = 1_8
    DO k=1,j-1
      bins_coeff = bins_coeff*INT(bins_tmp(k),8)
    END DO
    intgr_coeff = intgr_coeff - INT((kp(j)-1_4)*bins_coeff,8)
  END DO
  bins_coeff = 1_8
  DO j=1,i-1
    bins_coeff = bins_coeff*INT(bins_tmp(j),8)
  END DO
  kp(i) = 1 + INT(intgr_coeff/REAL(bins_coeff),4)
END DO
kp(1) = kp(1) - 1

! Transform bin coordinates to real values:
j = 0
DO i=1,SIZE(elements)
  IF (elements(i)) THEN
    j = j + 1
    array(i) = bounds(i,1) + kp(j)*bin_size(i)
  ELSE
    array(i) = -1.0
  END IF
END DO

DEALLOCATE(kp, help, bins_tmp)

END SUBROUTINE integerToArray

```

Appendix B

Algorithms for sorting and searching

B.1 Insertion sort

The insertion sort technique is the same as the one used for sorting cards: Pick out the second card and put it in order with the first card. Then pick out the third card and put it into the sequence among the first two. Continue until the last card has been picked out and inserted. In a random situation the algorithm scales as N^2 , but assuming an almost sorted array it can even reach $\sim N$. Generally, the insertion sort should not be used for $N \gtrsim 100$.

The following Fortran 95 code shows a general insertion sort algorithm implemented as a memory saving in-place sort, i.e., already sorted items are not stored in an additional temporary array.

```
SUBROUTINE insertionSort(array)

    IMPLICIT NONE
    INTEGER, DIMENSION(:), INTENT(inout) :: array
    INTEGER :: element
    INTEGER :: i, j

    DO i=2, SIZE(array)
        element = array(i)
        j=i
        DO WHILE (j>1 .AND. array(j-1)>element)
            array(j) = array(j-1)
            j = j - 1
        END DO
    END DO
```

```
        array(j) = element
    END DO
```

```
END SUBROUTINE insertionSort
```

B.2 Binary search

As a prerequisite, the array which is to be scanned, has to be sorted, usually in ascending order. The algorithm searches for the search key by repeatedly dividing the search interval in half. Initially the interval covers the whole array. If the item in the middle of the interval is larger than the key, the interval is narrowed to the lower half. Otherwise it is narrowed to the upper half. The procedure is continued until the value is found, or the interval is empty.

An example of the implementation of the algorithm in Fortran 95 is given below.

```
INTEGER FUNCTION binarySearch(key,array)

    ! The input array needs to be sorted in ascending order.

    IMPLICIT NONE
    INTEGER, INTENT(in)           :: key
    INTEGER, DIMENSION(:), INTENT(in) :: array
    INTEGER                       :: n, left, right, center

    n = SIZE(array)
    ! Return immediately, if the key (value to be searched for) is
    ! smaller or larger than the minimum or maximum values of
    ! the array:
    IF (key < array(1) .OR. key > array(n)) THEN
        binarySearch = -1
        RETURN
    END IF

    left = 1
    right = n
    DO WHILE (left <= right)
        center = (left+right)/2
        IF (key == array(center)) THEN
            binarySearch = center
```

```
        RETURN
    ELSE IF (key < array(center)) THEN
        right = center - 1
    ELSE IF (key > array(center)) THEN
        left = center + 1
    END IF
END DO

! The key could not be found in the array.
! Return a negative index:
binarySearch = -1

END FUNCTION binarySearch
```