

# Welcome to the

# *Comet/asteroid Orbit Determination and Ephemeris Software*

## Download Site and User's Manual

by [Jim Baer](#)

### Contents:

[Introduction](#)

[System Requirements](#)

[Download and Installation Instructions](#)

- [Java Runtime Environment](#)
- [CODES application and supporting files](#)
- [BC-405 asteroid ephemeris](#)

[How to Start \*\*CODES\*\*](#)

- [\(Windows 95/98/2000/NT/XP\)](#)
- [\(Solaris/Unix/Linux\)](#)
- [Mac with OS X](#)

[Background Documentation](#)

[Operating Instructions](#)

- [Level One Menu - Creating, Editing, or Deleting a minor planet](#)

- [Create a new minor planet](#)
  - [Edit an existing minor planet](#)
  - [Delete an existing minor planet](#)
- [Level Two Menu](#)
  - [Designate the minor planet as a comet or asteroid](#)
  - [Add a new observation](#)
    - [Add an optical observation](#)
    - [Add a radar observation](#)
    - [Read observations from an MPEC \(Minor Planet Electronic Circular\) or MPCOBS file](#)
    - [Read observations from a NEODYS file](#)
  - [Review/Delete Existing Observations](#)
  - [Compute/Evaluate/Propagate the orbit of this minor planet](#)
    - [Compute an initial two-body orbit](#)
    - [Compute an n-body best-fit orbit](#)
    - [Sample valid orbits: Observational Monte Carlo](#)
    - [Sample valid orbits: Statistical Ranging](#)
    - [Compute the observational residuals resulting from the current orbit](#)
    - [Propagate the current orbit to a new epoch](#)
  - [Display Current Orbit](#)
  - [Manually specify an orbit for this minor planet](#)
  - [Import a minor planet from the MPCORBcr or COMET catalogs](#)
  - [Compare observations to positions of known minor planets](#)
  - [Generate an optical ephemeris](#)
  - [Check for collisions and/or near-misses](#)
  - [Change the number of perturbing bodies for integrator](#)
  - [Specify the radius/visual brightness parameters](#)

## [Where to find data for use in CODES](#)

## [Notes on numerical integration and the accuracy of CODES](#)

- [Numerical Integration](#)
- [Accuracy of CODES](#)

## [Data Input Errors](#)

## [Release Information](#)

## Introduction

In support of current astronomical surveys for minor planets, CODES combines an n-body numerical integrator with a Graphical User Interface to provide the following capabilities:

- calculate the state vector/orbital elements (with covariances) of a comet or asteroid, based on up to 3000 optical and radar observations;
- identify known minor planets that most closely fit the observations of a comet or asteroid;
- calculate topocentric or geocentric ephemerides, based either on calculated, user-specified, or imported MPC state vector/orbital elements;
- linear and non-linear analysis of collisions/near-misses between a minor planet and the nine major planets (plus Earth's Moon) through 2200;
- account for cometary thrusting (if applicable), solar radiation pressure, solar oblateness, and gravitational perturbations (including relativistic effects) from the Sun, nine planets, Earth's Moon, and up to 300 asteroids.

## System Requirements

- approximately 150 MB of disk storage
- Java 1.4.1 (or higher) runtime environment (see [Download and Installation Instructions](#))
- suggested RAM - 1 GB or greater

- narrowband or broadband internet access (for download of observation files and updated CODES data files)
- To run CODES on a Macintosh requires Mac OS X version 10.2.3 or later. Mac OS X Panther initially shipped with installations of J2SE 1.3.1 and 1.4.1, and Java 1.4.2 is now available for Mac OS X Panther v10.3.1. Mac OS X Jaguar initially shipped with J2SE 1.3.1 only. Java 1.4.1 Update 1, based on the J2SE 1.4.1\_01 SDK, is available to Jaguar users through Software Update. Any issues with Java on the Macintosh should be referred to Apple's FAQ at [http://developer.apple.com/java/faq/#gen\\_1](http://developer.apple.com/java/faq/#gen_1)

## Download and Installation Instructions

- Java Runtime Environment

The CODES application is written in Pure Java; that is, the source code contains no commands specific to any particular operating system. As a result, compiled Java source code (known as "bytecode") can be run on **any** operating system.

In order to achieve Java's universality, Sun Microsystems has created a Java Virtual Machine (VM) to run on top of each popular operating system. The VMs provide a standard reference platform for the bytecode, translating it into processor-specific assembly code, which is then executed.

Thus, in order to run a compiled Java application such as **CODES**, a user must install the Java Runtime Environment (JRE) specific to their operating system, consisting of the VM and a "Just-in-Time" compiler that substantially accelerates bytecode processing.

The Java Runtime Environment, and instructions for installation, can be downloaded from [Sun's web site](#).

- CODES application and supporting files

To download and install CODES, the user should

1. Create a new folder (Windows and Mac OS X users) or directory (Solaris, Unix or Linux users) named "codes". (Note: For Mac OS X users, this new folder should lie within the "Applications" folder)
2. Download the following files from this web site to the "codes" folder/directory:
  - [CelestialBody](#) (save to disk as "CelestialBody.class") - last modified **6 November 2007**
  - [SolarSystemBody](#) (save to disk as "SolarSystemBody.class") - last modified **6 November 2007**
  - [MinorPlanet](#) (save to disk as "MinorPlanet.class") - last modified **6 November 2007**

- [GUI](#) (save to disk as "GUI.class") - last modified 12 November 2007
- [GUI\\$1](#) (save to disk as "GUI\$1.class") - last modified 12 November 2007
- [Graphing](#) (save to disk as "Graphing.class") - last modified 6 November 2007
- [Graphing\\$1](#) (save to disk as "Graphing\$1.class") - last modified 6 November 2007
- [Graphing\\$2](#) (save to disk as "Graphing\$2.class") - last modified 6 November 2007
- [ObsCode](#) (save to disk as "ObsCode.txt") - last modified 6 November 2007
- [CelestialBody source code \(optional\)](#) (save to disk as "CelestialBody.java") - last modified 6 November 2007
- [SolarSystemBody source code \(optional\)](#) (save to disk as "SolarSystemBody.java") - last modified 6 November 2007
- [MinorPlanet source code \(optional\)](#) (save to disk as "MinorPlanet.java") - last modified 6 November 2007
- [GUI source code \(optional\)](#) (save to disk as "GUI.java") - last modified 12 November 2007
- [Graphing source code \(optional\)](#) (save to disk as "Graphing.java") - last modified 6 November 2007
- [cometold](#) (save to disk as (cometold.txt) - last modified 6 November 2007

**Important Note:** As the Minor Planet Center certifies new contributing observatories, they must be added to the "ObsCode" file to support accurate processing of MPC and NEODyS observation files; users are therefore advised to return to this web site (at least) monthly for updated versions of the "ObsCode" file.

**Important Note:** I would strongly urge users of this most recent version of CODES to install and use the Java 1.4.1 (or later) runtime environment (JRE). Due to a recent change in the Java compiler, users with the Java 1.3 (or earlier) runtime environment may experience difficulty running the bytecode provided above (which was compiled in a 1.4.1 environment). In short, the source code is still perfectly valid, but the method by which font-related source code commands are compiled into bytecode has changed in Java 1.4.1; as a result, the corresponding 1.4.1 bytecode would not be interpreted correctly in a 1.3 (or earlier) environment. Users may also elect to recompile the source code provided above in their 1.3 (or earlier) JDK environment; the classes should be compiled in the following "top-down" order: CelestialBody, SolarSystemBody, MinorPlanet, and finally, GUI.

**Important note:** To support accurate processing of MPC and NEODyS observation files, CODES accounts for the number of leap seconds at the time of each observation. Additional leap seconds are added only as irregular changes in the Earth's rotation warrant; moreover, by convention, any additional leap seconds are added only on the 30'th of June or the 31'st of December of a given year. Users are therefore advised to return to this web site for updated versions of the CelestialBody file whenever additional leap seconds are announced. The

**current version of CODES supports the leap second to be added 31 December 2005.**

**Important note for Solaris/Unix/Linux users: The file "GUI\$1.class" may be renamed "GUI\_1.class" during download. If this occurs, please rename it as "GUI\$1.class" so the Java Runtime Environment can find it.**

3. Download the following DE405 planetary ephemeris files (ASCII versions) from the [JPL web site](#) to the "codes" folder/directory:

- **ASCP1800.405** (save to disk as "ASCP1800.txt")
- **ASCP1820.405** (save to disk as "ASCP1820.txt")
- **ASCP1840.405** (save to disk as "ASCP1840.txt")
- **ASCP1860.405** (save to disk as "ASCP1860.txt")
- **ASCP1880.405** (save to disk as "ASCP1880.txt")
- **ASCP1900.405** (save to disk as "ASCP1900.txt")
- **ASCP1920.405** (save to disk as "ASCP1920.txt")
- **ASCP1940.405** (save to disk as "ASCP1940.txt")
- **ASCP1960.405** (save to disk as "ASCP1960.txt")
- **ASCP1980.405** (save to disk as "ASCP1980.txt")
- **ASCP2000.405** (save to disk as "ASCP2000.txt")
- **ASCP2020.405** (save to disk as "ASCP2020.txt")
- **ASCP2040.405** (save to disk as "ASCP2040.txt")
- **ASCP2060.405** (save to disk as "ASCP2060.txt")
- **ASCP2080.405** (save to disk as "ASCP2080.txt")
- **ASCP2100.405** (save to disk as "ASCP2100.txt")
- **ASCP2120.405** (save to disk as "ASCP2120.txt")
- **ASCP2140.405** (save to disk as "ASCP2140.txt")
- **ASCP2160.405** (save to disk as "ASCP2160.txt")
- **ASCP2180.405** (save to disk as "ASCP2180.txt")

4. Download the following MPC orbital element files and save them to the "codes" folder/directory, renaming them as listed:

- [MPCORBcr.DAT](#)
- [COMET.DAT](#)

Note: If the user chooses to download the [zipped version of MPCORBcr](#), simply unzip the file and save it to the "codes" folder/directory, renaming it as "MPCORBcr.DAT".

Note: With some or all versions of Macintosh OS X, CODES may not recognize MPCORBcr.DAT and COMET.DAT as text-files, and may be unable to access the files after download.

This will result in failure to import a minor planet from the MPCORBcr or COMET catalogs, and false negative responses when comparing observations to positions of known minor planets. To correct this problem and ensure proper operation of CODES with these files, please perform the following check/operation after downloading the two files: Select the file MPCORBcr.DAT by clicking on it once. Select the "File"-menu, "Get Info"-command. In the window that opens, click on the triangle "Open with:" and select the application "TextEdit" on the drop-down menu. Perform the same operation with the COMET.DAT file. **IMPORTANT:** With a Macintosh, it is necessary to repeat the above procedure after every new download of the MPCORBcr.DAT and COMET.DAT files.

5. Download and unzip the text version of the [BC-405 Asteroid Ephemeris](#) (see next item for details) to the "codes" folder/directory, renaming it as "asteroid\_ephemeris.txt"

6. (Optional, but strongly recommended) Download the current Earth Orientation Parameter file "EOPC04" from the web site of the [International Earth Rotational Service](#), and save it to the "codes" folder/directory as text file "eopc04.txt".

- [BC-405 Asteroid Ephemeris](#)

The new BC-405 Asteroid Ephemeris allows CODES to calculate perturbations based on integrated state vectors of the same 300 asteroids used in DE-405. This should increase the accuracy of orbit determinations and impact assessments, especially over long timeframes.

Additionally, BC-405 can be used separately, or integrated into other applications. The format is explained [here](#), and the asteroid masses used in the ephemeris (as well as the order in which the asteroids appear in each file record) is [here](#).

My sincere thanks to Gareth Williams of the IAU Minor Planet Center for hosting the [zipped text version](#) of the ephemeris. A version in SPK Type 5 format is also available [here](#), thanks to Charles Acton and Steve Chesley of JPL. \_

## How to Start CODES

- (Windows 95/98/2000/NT/XP)

1. From the Start button, select Programs -> MS-DOS Prompt
2. Type "cd c:\codes" to change to the "codes" folder, which contains the above downloaded files
3. Type "java -Xmx600m GUI" (Note: The "-Xmx600m" portion of the command increases maximum heap space from 64 MB to 600 MB)
4. Begin at the **CODES** Welcome Screen

- (Solaris/Unix/Linux)

1. Change to the "codes" directory, which contains the above downloaded files
  2. Type "java -Xmx600m GUI" (Note: The "-Xmx600m" portion of the command increases maximum heap space from 64 MB to 600 MB)
  3. Begin at the **CODES** Welcome Screen
- (Mac with OS X)
    1. Launch the "Terminal" utility (available in Applications:Utilities)
    2. Type "cd/applications/codes/" to switch to the codes folder, which contains the above downloaded files
    3. Type "java -Xmx600m GUI" (Note: The "-Xmx600m" portion of the command increases maximum heap space from 64 MB to 600 MB. Smaller heap sizes may be used, but only if the orbit determination and/or nonlinear collision analysis functions are not used.)
    4. Begin at the **CODES** Welcome screen

Note: When CODES is run for the first time, users may notice the following error message: "Error -- java.io.FileNotFoundException: minor\_planet\_list (No such file or directory)". You should nevertheless be able to proceed. The "minor\_planet\_list" normally contains the list of comets and asteroids that have been added to CODES; as such, when the user hits "Begin" and enters the Level One Menu, CODES will try to read the file so as to provide the user with the list of available minor planet files to edit (or delete). Since none yet exist, the error message results. Once the user creates a minor planet, the "minor\_planet\_list" file will be created, and the error message should not recur.

Note: My sincere thanks to Christian Kjaernet for determining (and sharing) the Mac OS X start-up procedure.

## Background Documentation

The [requirements document](#) and [dynamical model](#) are also available. I would recommend reading them before proceeding further, as they give an overview of the capabilities of the software, and the design of its Graphical User Interface.

## Operating Instructions

CODES was designed and written as an object-oriented application. The general idea, therefore, is to create an instance of the Minor Planet class (e.g. asteroid 1997 XF11), specify some attributes (e.g. optical and/or radar observations), and execute various functions or operations (e.g. compute initial and best-fit orbit, check for collision/near-miss) that will generate the values of additional attributes (state vector/orbital elements, absolute magnitude, estimated radius). Following valid data entry or the successful completion of operations, the file for the subject minor planet, containing the values of all attributes, is saved to disk.



The application has a two-tiered menu.

The Level One Menu allows the user to select whether to

- create new instances of minor planets;
- edit the attributes of, or conduct operations on, existing minor planets; or
- delete existing minor planets.

The Level Two Menu provides the graphical interface allowing the user to specify the attributes of a particular minor planet and conduct various operations. From any subsequent screen, it is possible to cancel (or end) the current function and return to the Level Two Menu.

Begin at the Welcome screen by clicking on "Begin"

- Level One Menu - Creating, Editing, or Deleting a minor planet
  - Create a new minor planet

Simply enter the desired name in the text field and press "Create".

The name you specify will become the name of the data file storing that minor planet's attributes (observations, orbital elements, etc.). Thus, there are Java language restrictions on permissible characters. All alphanumeric characters are permitted, but slashes ("/") and empty spaces (" "), for example, are not permitted. Once the user clicks on "Create", CODES will examine the proposed name and remove any illegal characters. My general guidance, therefore, would be that the user should feel free to use whatever name, number, or provisional designation is desired or appropriate; CODES will remove any illegal characters as needed.

The user will then have the option of returning to the [Level One Menu](#), or continuing on to edit the newly created minor planet from the [Level Two Menu](#).

The advanced user also has the option of executing an "Expedited Comparison". CODES will assume that both an mpec.txt observation file for this minor planet and a current copy of MPCORBcr.DAT are available in the "codes" folder; CODES will [process the observations](#), add them to the minor planet's data file, and [determine which known single-opposition NEAs best match these observations](#) (using medium-accuracy numerical integration). Note that this option should be used with care, since the absence of the appropriate version of either file from the "codes" folder will result in

a crash.

- Edit an existing minor planet

This option allows the user to select a particular pre-existing minor planet, and add attribute data (i.e., specify optical or radar observations) or perform operations (e.g., compute the orbit, check for collisions/near-misses).

Simply select the desired entry from the drop-down menu and press "Edit". This will take the user to the [Level Two Menu](#).

- Delete an existing minor planet

Simply select the desired minor planet from the drop-down menu and press "Delete". The file associated with the selected minor planet is deleted.

The user may then return to the [Level One Menu](#).

- Level Two Menu - Specifying attributes of, and executing operations on, the subject minor planet

- Designate the minor planet as a comet or asteroid

For several operations (e.g., processing visual magnitude data, looking for known bodies whose positions agree with the observations of the subject minor planet), it is critical that CODES understand whether the subject minor planet is a comet or an asteroid.

Thus, **this designation should be made immediately upon creating a new minor planet.**

From the Level Two Menu, simply click on "Designate"; on the resulting frame, select whether the minor planet is a comet or asteroid, then click on "Submit".

The designation being made, CODES will save the data to disk and return to the [Level Two Menu](#).

- Add a new observation

From the Level Two Menu, simply click on "Add". From the resulting frame, the user can then select from the following options:

■ Add an optical observation

Simply click on "Optical". The user will then encounter up to three successive frames:

The first frame allows the user to specify the UTC date of the optical observation, the astrometric right ascension and declination (with uncertainties), the visual magnitude, and the fundamental catalog/epoch to which the observation refers.

**Note:** CODES supports observations and predictions for the range 1800-2200.

The second frame allows the user to specify either the MPC observatory code, or the geodetic latitude, east longitude, and altitude, from which the observation was made. Additionally, the user may check a box telling CODES to generate default Earth Orientation Parameters; the third frame would then be skipped.

**Note:** For typical optical observations, where uncertainties are on the order of 0.1 arc seconds or greater, the default EOPs generated by CODES are sufficient. The option to specify EOPs for optical observations will allow future support of more precise astrometric techniques.

The third frame allows the user to specify the Earth Orientation Parameters (EOPs) at the time of the observation. EOPs allow calculation of the ultra-precise location of the observer. These EOPs include the x and y coordinates of the Celestial Ephemeris Pole, the values of UT1-UTC, and the Celestial Pole Offsets dPsi and dEpsilon. (Note that these EOPs are required for four dates surrounding the observation; Besselian interpolation is then used to approximate the values of the EOPs at the precise observation time.) Additionally, the user must specify the number of leap seconds and the Earth's angular velocity at the time of observation.

**IMPORTANT:** CODES now supports use of the International Earth Rotational Service's "EOPC04" tables. Simply download the current "[Series from 1962 to the current week](#)" file, and save it in the "codes" folder as "eopc04.txt". If the file is present, and the observation date falls within the table's span, CODES will automatically read the relevant EOPs, and skip the third frame. If the file is not present, or if the observation date is not covered by the table, CODES will either generate default EOPs (if this option is checked), or proceed to the third frame for manual entry of the EOPs.

**IMPORTANT:** Do not leave any data fields blank! This can cause a run-time error.

**IMPORTANT:** From either of the three input frames, it is possible to cancel the input function and return to the [Level Two Menu](#); the new (and incomplete) observation will be deleted.

When complete, CODES will save the data to disk and return to the [Level Two Menu](#).

■ Add a radar observation

Simply click on "Radar". The user will then encounter three successive frames:

The first frame allows the user to specify the UTC time of observation, whether this observation is a delay or a doppler measurement, the measurement and its uncertainty, and the transmitter frequency.

**Note:** CODES supports observations and predictions for the range 1800-2200.

The second frame allows the user to specify the location of both the transmitter site and the receiver site. In each case, the user may either select one of several common radar sites (in which case the lat/long/alt data fields may safely be left blank), or select "other" and specify the geodetic latitude, east longitude, and altitude of the transmitter/receiver.

The third frame allows the user to specify the Earth Orientation Parameters (EOPs) at the time of the observation. EOPs allow calculation of the ultra-precise location of the observer. These EOPs include the x and y coordinates of the Celestial Ephemeris Pole, the values of UT1-UTC, and the Celestial Pole Offsets dPsi and dEpsilon. (Note that these EOPs are required for four dates surrounding the observation; Besselian interpolation is then used to approximate the values of the EOPs at the precise observation time.) Additionally, the user must specify the number of leap seconds and the Earth's angular velocity at the time of observation.

**IMPORTANT:** CODES now supports use of the International Earth Rotational Service's "EOPC04" tables. Simply download the current "[Series from 1962 to the current week](#)" file, and save it in the "codes" folder as "eopc04.txt". If the file is present, and the observation date falls within the table's span, CODES will automatically read the relevant EOPs, and skip the third frame. If the file is not present, or if the observation date is not covered by the table, CODES will proceed to the third frame for manual entry of the EOPs.

**Note:** For radar data, accuracies of 0.1 microsecond or 30 meters are typical. My sensitivity analyses indicate that 1 meter accuracy could be supported without interpolating the published EOPs. However, the state-of-the-art is

advancing so rapidly that I feel it prudent to accommodate future radar technology (up to 0.1 meter accuracy); thus, second-order interpolation of the EOPs is used.

**IMPORTANT:** Unless otherwise noted, do not leave any data fields blank! This can cause a run-time error.

**IMPORTANT:** From either of the three input frames, it is possible to cancel the input function and return to the [Level Two Menu](#); the new (and incomplete) observation will be deleted.

When complete, CODES will save the data to disk and return to the [Level Two Menu](#).

- Read observations from an MPEC (Minor Planet Electronic Circular) or MPCOBS file

The Minor Planet Center (MPC) provides free [web-based electronic circulars](#) containing formatted tables of new observations of recently discovered and high-interest minor planets. The MPC also provides an Extended Computer Service called MPCOBS which enables users to extract files containing all observations of a specific minor planet.

With some help from the user, CODES can batch-process the entire set of observations contained in each of these files, thus avoiding the need for manual input.

Simply click on "MPC". The next frame will ask the user to save the MPEC or MPCOBS file (from the web browser or text editor) as "mpec.txt" in the "codes" folder, using file-type "text file" or "text-only".

**IMPORTANT:** *Some MPECs include observations of many minor planets. In such cases, the user should edit out the observations of other minor planets, leaving only the observations of the desired minor planet.*

Note: Users will often want to refine computed orbits (and collision analyses) using additional observations as they become available. However, MPECs and MPC files list all observations of the subject minor planet, including those which CODES has (presumably) previously processed. To avoid the creation of duplicate entries, CODES checks each observation as it is being processed; if an observation with the same date, observation location, and measurement already exists, CODES will ignore this observation and skip to the next.

When ready, simply click on "Process".

The MPC format allows CODES to extract the following data for each observation:

- UTC time of observation (receive-time for radar observations)
- observational measurement plus uncertainty (right ascension, declination, and visual magnitude for optical observations, delay and/or doppler for radar observations)
- Observer location (including MPC observatory code for fixed optical observations, lat/long/alt for roving observers, x/y/z for satellite observations, and both transmitter and receiver MPC observatory codes for radar observations)

CODES then either reads or generates default Earth Orientation Parameters and processes the observation.

**IMPORTANT:** CODES now supports use of the International Earth Rotational Service's "EOPC04" tables. Simply download the current "[Series from 1962 to the current week](#)" file, and save it in the "codes" folder as "eopc04.txt". If the file is present, and the observation date falls within the table's span, CODES will automatically read the relevant EOPs and use them in determining the observer's position. If the file is not present, or if the observation date is not covered by the table, CODES will generate default EOPs.

**IMPORTANT:** At present, the default EOPs determine observer position to within about 1 km. While this accuracy is sufficient for current optical observations whose precision average about 0.1 arc seconds (and irrelevant for satellite observations, whose observer positions are read directly from the observation data record), it will not fully support highly-precise radar delay and/or doppler observations. The user is therefore strongly urged to ensure the current "eopc04.txt" file is available for processing any radar observations.

**IMPORTANT:** The MPC is constantly certifying new contributing observatories. If a user were to process an MPEC or MPCOBS file containing observations from a new observatory that is not on the CODES look-up table, the quality of derived data would be severely degraded. The user is therefore advised to periodically either check this web site or [contact me](#) for updated copies of the "ObsCode.txt" file. New versions of the file should overwrite the existing version in the "codes" folder/directory.

**Note:** CODES supports observations and predictions for the range 1800-2200.

When complete, CODES will save the data to disk and return to the [Level Two Menu](#).

- Read observations from a NEODYS file

The Near-Earth Objects Dynamic Site (NEODyS) provides web-based data on every known Near-Earth Object (NEO). Start at the [search page](#), and enter the name, number, or designation of an NEO; a web page is dynamically created providing the user with data on that NEO.

Near the bottom of the web page, under "Observational Information:", click on the "ASCII file" link. The user will be provided with a formatted table containing all known observations of the NEO. With some help from the user, CODES can batch-process the entire set of observations, thus avoiding the need for manual input.

Simply click on "NEODyS". The next frame will ask the user to save the NEODyS file from the web browser as "neodys.txt" in the "codes" folder, using file-type "text file" or "text-only".

Note: Users will often want to refine computed orbits (and collision analyses) using additional observations as they become available. However, NEODyS files list all observations of the subject minor planet, including those which CODES has (presumably) previously processed. To avoid the creation of duplicate entries, CODES checks each observation as it is being processed; if an observation with the same date, observation location, and measurement already exists, CODES will ignore this observation and skip to the next.

When ready, simply click on "Process".

The NEODyS format allows CODES to extract the following data for each observation:

- UTC time of observation (receive-time for radar observations)
- observational measurement plus uncertainty (right ascension, declination, and visual magnitude for optical observations, delay or doppler for radar observations)
- Observer location (including MPC observatory code for fixed optical observations, and both transmitter and receiver MPC observatory codes for radar observations)

CODES then either reads or generates default Earth Orientation Parameters and processes the observation.

**IMPORTANT:** CODES now supports use of the International Earth Rotational Service's "EOPC04" tables. Simply download the current "[Series from 1962 to the current week](#)" file, and save it in the "codes" folder as "eopc04.txt". If the file is present, and the observation date falls within the table's span, CODES will automatically read the relevant EOPs and use them in determining



the observer's position. If the file is not present, or if the observation date is not covered by the table, CODES will generate default EOPs.

**IMPORTANT:** At present, the default EOPs determine observer position to within about 1 km. While this accuracy is sufficient for current optical observations whose precision average about 0.1 arc seconds (and irrelevant for satellite observations, whose observer positions are read directly from the observation data record), it will not fully support highly-precise radar delay and/or doppler observations. The user is therefore strongly urged to ensure the current "eopc04.txt" file is available for processing any radar observations.

**IMPORTANT:** The MPC is constantly certifying new contributing observatories. If a user were to process a NEODYs file containing observations from a new observatory that is not on the CODES look-up table, the quality of derived data would be severely degraded. The user is therefore advised to periodically either check this web site or [contact me](#) for updated copies of the "ObsCode.txt" file. New versions of the file should overwrite the existing version in the "codes" folder.

**IMPORTANT:** There are currently no "roving observer" or satellite-based observations of Near-Earth Objects in the NEODYs database. Should they appear in the future, I believe "roving observer" observations would be listed in a NEODYs file with observatory code "247", while satellite observations would be listed with observatory code "248" (Hipparcos), "249" (SOHO), or "250" (Hubble Space Telescope). Since there is no fixed position associated with these observatories, there is no way to accurately calculate the observer position with the data provided. The user is therefore urged to get the data for these observations from an MPEC or MPCOBS file.

**Note:** CODES supports observations and predictions for the range 1800-2200.

When complete, CODES will save the data to disk and return to the [Level Two Menu](#).

- Review/Delete Existing Observations

This option permits the user to review the observations that exist for the given minor planet, and to exclude those observations (if desired) which are found to have exceptionally high residuals.

From the [Level Two Menu](#), simply click "Rev/Del". From the resulting frame, the user can review observations five-at-a-time. The user can delete one or more of the displayed observations by checking the box to the left of the observation(s), then clicking "Delete Checked". When more than five observations exist, the user can click on the "Next 5" and "Prev 5" buttons to cycle through the entire list.



When done, simply click on "Done"; CODES will save the data to disk and return to the [Level Two Menu](#).

- Compute/Evaluate/Propagate the orbit of this minor planet

Simply click on "Compute". From the resulting menu, select one of the four options:

- Compute an initial two-body orbit

Once at least three observations exist for a minor planet, it is possible to calculate an initial orbit using two-body mechanics (This initial orbit can later be refined by using integrated n-body mechanics to calculate a best-fit least-squares orbit - see "Compute an n-body best-fit orbit"). If only two observations exist, an orbit can be estimated based on additional user-inputs.

Simply click "Initial". On the resulting frame, use the drop-down menus to select which three observations to use in calculating the initial orbit. (Note: Only the first two selections will be used in Herget's method.)

Knowing which observations to use requires some judgment. The objective is to cover a significant arc of the full orbit. For distant minor planets such as Trans-Neptunian Objects, it may be necessary to have an arc of data covering tens or even hundreds of days to get a good initial orbit; conversely, a Near-Earth Object or a minor planet in the main asteroid belt may require an arc of just a few days. My best advice is to experiment. If all of the initial orbit methods diverge or give absurdly high residuals, pick another set of observations.

Once the user has made the selections, simply press "Submit". From the resulting menu, the user must now select which method of initial orbit determination to try.

I have attempted to make these methods as robust as possible. However, each algorithm has inherent limits.

- The Gauss method is clearly the very best, and it is always my first choice. If the iterative algorithm is successful, the calculated orbit will contain the three observations; thus, you can immediately accept the orbit (and skip differential correction, since the residuals will be near zero). However, Gauss is sometimes less successful for Near-Earth Objects.

Simply click on "Gauss". The resulting frame will display the calculated state vector and derived orbital elements for the minor planet at the TDB time of the first selected observation. The two-body residuals for each

selected observation will be displayed, as well as a message warning if the iterative algorithm has diverged (in which case the first-approximation to the orbital elements/state vector is displayed).

The user will have several options.

1. If satisfied with the initial orbit, simply click on the "Select" button next to "Accept This Orbit"; the initial orbit will be saved to disk, and CODES will return to the [Level Two Menu](#).
2. If the residuals are non-zero, the user may elect to click on the "Select" button next to "Apply Differential Correction". The same frame will appear, displaying the resulting state vector and derived orbital elements for the minor planet at the TDB time of the first selected observation. The two-body residuals for each selected observation will be displayed, as well as a message warning if the differential correction algorithm has diverged (in which case the original Gauss method orbital elements/state vector is displayed). When differential correction converges successfully to zero residuals, the user can accept the orbit. But when differential correction diverges, it's usually necessary to [try another method](#), or to [select another set of three observations](#).
3. If the Gauss method and/or differential correction have diverged, the user may elect to click on the "Select" button next to "Try Another Method". In this case, the three selected observations are retained, and CODES returns to the menu of methods of initial orbit determination.
4. If none of the three methods of initial orbit determination has converged successfully, the user may elect to click on the "Select" button next to "Try a Different Set of Three Observations". In this case, CODES returns to the frame with the drop-down menus to [select a new set of three observations](#), followed by the frame to select one of the three methods of initial orbit determination.
5. If the user wishes to exit the initial orbit determination process, simply click on the "Select" button next to "Quit"; no initial orbit will be saved, and CODES will return to the [Level Two Menu](#).

- The conditioned Gauss method requires the user to initialize the algorithm by providing a guess for the semi-major axis. Since it is mostly used for Near-Earth Objects, a guess of "1.0" is a good

starting point.

Simply click "conditioned Gauss". From the next frame, enter a guess for the semi-major-axis and click on "Submit". The resulting frame will display the calculated state vector and derived orbital elements for the minor planet at the TDB time of the first selected observation. The two-body residuals for each selected observation will be displayed, as well as a message warning if the iterative algorithm has diverged (in which case the first-approximation to the orbital elements/state vector is displayed).

The user will have the [same options as in the case of the Gauss Method](#), with one addition:

1. If the resulting residuals are too high and differential correction has diverged, the user may elect to click on the "Select" button next to "Try Another Value of Semi-Major Axis". In this case, CODES returns to the frame where the user can provide another guess.

If the user's guess for the semi-major axis is close to the correct value, the calculated orbit will contain the first and second selected observations, with a small residual (approx. 10 arc seconds or less) on the third selected observation. Use of differential correction usually improves the fit, and allows you to accept the orbit.

If the user's guess for the semi-major axis is not close to the correct value, the residual on the third selected observation will be large (approx. 10 arc seconds or greater). In this case, differential correction may diverge. Try different values for the semi-major axis, attempting to minimize the residual on the third selected observation.

- The Laplace method is usually the method of last resort; it is sometimes useful for closely-spaced observations (as is often the case with TNOs). Differential correction will always be required here, since the calculated orbit will rarely include any of the three selected observations.

Simply click "Laplace method". The resulting frame will display the calculated state vector and derived orbital elements for the minor planet at the TDB time of the second selected observation. The two-body residuals for each selected observation will be displayed, as well as a message warning if the iterative algorithm has diverged (in which case the first-approximation to the orbital elements/state vector is displayed).

The user will then have the [same options as in the case of the Gauss Method](#).

- Herget's method requires the user to initialize the algorithm by providing guesses for the slant range at the time of each selected observation. Testing has shown that multiple solutions may exist for closely-spaced observations, and for small slant ranges; my current recommendation would be to apply this method for objects no closer than the main-belt.

Simply click "Herget's method". The algorithm will create an estimate of the state vector at the time of the first selected observation. If the user specifies, least-squares will then be applied in an attempt to refine this state vector by minimizing its residuals for all available observations. If least-squares converges, the resulting state vector will actually be an n-body best-fit orbit. If least-squares does not converge, the residuals could be quite large.

The resulting frame will display the calculated state vector and derived orbital elements for the minor planet at the TDB time of the first selected observation. The n-body RMS residuals will be displayed, as well as a message warning if least squares diverged (in which case the first-approximation to the orbital elements/state vector is displayed).

The user will then have the same options as in the case of the conditioned Gauss method, except that "Apply differential correction" is replaced by "Apply least squares", and "Try another value of Semi-Major Axis" is replaced by "Try another value of Slant Range".

**Important Note:** The set of initial orbital elements displayed by each of the above methods will depend upon whether the user has designated this minor planet as an asteroid or comet. It is therefore vitally important that the user [make the appropriate selection](#) before beginning this option.

#### ■ Compute an n-body best-fit orbit

This option allows the user to refine the existing orbit for the minor planet (either a calculated initial orbit, or an orbit specified by the user) in the least-squares sense, using numerical integration and n-body mechanics. The products are the best-fit state vector plus covariances (based on the observational uncertainties), the derived orbital elements plus covariances, the calculated residuals for each observation, the visual absolute magnitude and slope parameter (if apparent magnitude observations are available), and an order-of-magnitude estimate of the minor planet's radius (asteroids only, and subject to whether the visual absolute magnitude and slope parameter are available). Additionally, if the minor planet is potentially hazardous, an estimate of the Minimum Orbital Intersection Distance (MOID) will be provided.

To calculate the best-fit orbit, simply click "Best-Fit". From the resulting frame, select whether to account for cometary thrusting through the use of non-gravitational thrust parameters, and select whether to exclude any observations that exceed specified chi or residual thresholds; then click "Begin".

As noted, these calculations can take a great deal of time, depending on the computer processor, the total number of observations, the calendar arc of the observations, and the number of perturbing asteroids being used in the calculations.

Some important notes:

- This option uses n-body mechanics, accounting for
  1. gravitational perturbations (including relativistic effects) from the Sun, nine planets, Earth's Moon, and as many asteroids as the user has specified; as a result, it is very important that the user [select the desired number of perturbing bodies](#) from the Level Two Menu before attempting to calculate the best-fit orbit.
  2. solar radiation pressure (assuming a spherical shape)
  3. solar oblateness ( $J_2$  effects)
  4. cometary thrusting, if the user so specifies.
- Use of non-gravitational thrust parameters in best-fit orbit determination

The orbits of nearly all asteroids can be calculated with very high accuracy using a relativistic gravity model. However, the tails of comets are basically small rocket engines, propelling them approximately away from the Sun. This propulsive force must be modeled to enable calculation of cometary orbits with acceptably small residuals.

The non-gravitational thrust parameters include both a radial (A1) and a transverse (A2) component; these components can be calculated by CODES as part of the least-squares best-fit process if the user selects this option.

The parameters also include a perihelion time offset (DT), which is sometimes useful in further reducing residuals. However, DT is not determined as part of the best-fit orbit. Rather, the value of DT (default = 0) can be specified using the "[Specify minor planet's orbit](#)" option from the Level Two Menu; the user should then recalculate the observational residuals, and iteratively vary DT in increments of 5 days (up to values of +/- 60 days) to find the value of DT which minimizes the residuals.

If the user is studying an asteroid or a comet with few observations, the non-gravitational thrust parameters are not needed. But if the user is analyzing the motion of a comet over a long arc, the user may get much lower residuals by electing to solve for the best-fit orbit using these parameters. (Note also that the orbit of at least one Near-Earth Asteroid, which may be the stony nucleus of a now-expired comet, also requires the use of non-gravitational thrust parameters; thus, the user may elect to use these parameters regardless of whether they have designated the minor planet as a comet or asteroid.)

- Excluding observations with excessive chi or residuals

This option allows the user to set thresholds for chi, or for optical, delay, and doppler residuals. If enabled, the least-squares algorithm will first converge to a nominal solution using all observations; it will then continue for one or more additional iterations, this time excluding any observations exceeding the specified thresholds. The resulting final orbit will thus not be impacted by unreliable data (as defined by the user).

Note that the user could just as easily determine the best-fit orbit using all observations, manually delete those with excessive chi or residuals, then find the new best-fit orbit (sometimes having to repeat the process several times as changes in the best-fit orbit alternately include or exclude some observations); this option allows CODES to automate the process. Moreover, this option does not permanently delete excluded observations, but rather simply ignores them; this will allow these observations to be reconsidered in future runs, should additional observations change the best-fit orbit enough to make their departures permissible.

Any excluded observations are marked with an asterisk in the resulting frames and file output.

- Determining the best-fit orbit using observations spanning many years

The least-squares process often begins by using the initial two-body orbit to calculate the residual (in the sense of "observed - computed") for each observation. However, if the date of an observation differs from the epoch of the two-body orbit by two or more months, it is likely that the residual will be quite large. For observations spread over several decades, the problem becomes severe, and it is quite possible that the least-squares algorithm will diverge.

To address this problem, I suggest the following procedure:

1. First, enter only observations in a one-month period of time. Find the initial orbit, and then the best-fit orbit for these observations. Note the epoch.
2. Enter the observations from the one-year period of time around the epoch. Find the best-fit orbit for these observations.
3. Enter the rest of the observations, and find the final best-fit orbit.

(The suggested periods of time will vary depending on the perturbations experienced by the subject minor planet. For observations during close approaches by NEAs, the initial orbit may only span a few days, and the first attempt at a best-fit orbit may be limited to a week-long span.)

#### ■ Radar observations and the radius of the subject minor planet

The inherent precision of radar observations is such that CODES must account for the radius of the subject minor planet (estimating the "bounce point") when calculating the orbit of its center-of-mass. If apparent magnitude data is available, the least-squares algorithm can make an order-of-magnitude estimate of an asteroid's radius after the first iteration, and refine this estimate on subsequent iterations. But this is not currently possible for comets, and it is clearly not possible in the absence of apparent magnitude data. In these latter cases, it is therefore very important that the user [specify the radius](#) from the Level Two Menu prior to a best-fit calculation involving radar observations.

#### ■ Brightness parameters and asteroid radius

When the best-fit orbit of the subject minor planet is calculated, CODES uses apparent magnitude data (if available) to estimate (using different algorithms, depending on whether the object is a comet or asteroid) the absolute magnitude and slope parameter. Thus, it is very important that the user [select whether the minor planet is an asteroid or comet](#) before attempting the calculate the best-fit orbit.

For asteroids, the slope parameter is subsequently used to estimate the albedo, itself resulting in an order-of-magnitude estimate of the radius.

- **Important Note:** The set of best-fit orbital elements displayed will depend upon whether the user has designated this minor planet as an asteroid or comet. It is therefore vitally important that the user [make the appropriate selection](#) before beginning this option.

When the best-fit calculations are complete, the new orbit is saved to disk, and the results are summarized in the file "(minor planet name)\_best\_fit\_output.txt" in the "codes" folder.

The resulting frames successively display the residuals for each observation, the estimated physical properties of the minor planet (deduced from apparent magnitude data, if available), and the calculated state vector and orbital elements. To return to the [Level Two Menu](#) at any point, simply press "Done".

## § Sample valid orbits: Observational Monte Carlo

This feature has two primary purposes:

- § to aid in the recovery (and precovery) of minor planets for which a convergent least-squares solution exists, but where the uncertainty in future position is high;
- § to identify and evaluate multiple orbit solutions.

Upon selecting this option, CODES begins calculating 100 variant orbital solutions, based on adding small random uncertainties to each of the known observations and using least-squares to determine the resulting state vectors. In essence, these variant orbits represent "virtual solutions" that lie within 3-sigma in observation space of the nominal solution.

Currently, a variant orbit is discarded unless it satisfies the following:

- § the resulting least-squares solution is convergent;
- § the optical rms residual is less than three times that of the nominal solution;
- § the resulting orbit is prograde, and either elliptical or parabolic.

Once the surviving variant orbits are complete, CODES presents the user with the option of viewing any of five scatterplots:

- § q vs. e
- § q vs. i
- § q vs. w



§ q vs. omega

§ q vs. M

In these plots,

§ variant orbits for which the optical rms solution is between 0 and 1 times that of the nominal solution are **blue**

§ variant orbits for which the optical rms solution is between 1 and 2 times that of the nominal solution are **green**

§ variant orbits for which the optical rms solution is between 2 and 3 times that of the nominal solution are **red**

These scatterplots can help identify multiple orbital solutions; the color of the variant orbits corresponding to each solution can help the user judge which solution is more likely to realistically fit the known observations.

Having generated these variant orbits, the user can also [create ephemeris scatterplots](#), which would allow the user to define the area in which a minor planet might appear at given times; the color of the variant orbits corresponding to each solution can help the user further narrow the likely search region, and hopefully aid in recovery or precovery.

**Important Note:** This option should only be applied in cases where a convergent least-squares orbit has been calculated. Cases in which no convergent least-squares solution exists must be handled using Statistical Ranging, which will be an upcoming feature of CODES.

§ Sample valid orbits: Statistical Ranging

One of the great frustrations of orbit determination is that our most elegant initial orbit methods can sometimes fail to converge, especially for closely-spaced observations. And although the methods of Herget and Väisälä can often generate initial orbits in such cases which are consistent with two of the observations, such methods require significant assumptions as to the geometry of the orbit, they may not satisfy the other available observations, and they do not provide a thorough sampling of the orbital solution space. Lacking a definitive initial orbit, recovery (much less precovery) becomes difficult. Finally, even if an initial orbit is obtained, the method of least-squares explicitly assumes that linearity applies; in many interesting cases, this approximation is invalid.

By contrast, Statistical Ranging (SR) is essentially a brute force computational

search for any and all orbits satisfying a given set of observations. Linearity is not assumed; in fact, this method is ideal for generating ephemerides and collision analyses in highly non-linear circumstances, and for mapping the full orbital solution space in ambiguous cases.

The potential precovery/recovery applications of this new algorithm are clearly very significant. And the experience of AL00667 (alias 2004 AS1) has (in retrospect) demonstrated SR's ability to evaluate the near-term collision risk of newly-discovered NEAs.

Like the previous method, the Statistical Ranging algorithm consists of adding small random uncertainties to each of the known observations, and generating a variant trajectory which satisfies these "noisy" observations. But the similarity to Observational Monte Carlo ends there. (Note: There are at least two qualitatively different implementations of Statistical Ranging; the version implemented here is largely due to Virtanen and Muinonen.)

Once (Gaussian) noise has been added to the observations, the SR algorithm generates two random numbers. The first random number is used to create the slant range at the time of the first observation, based on user-specified minimum and maximum slant ranges. The second random number is used to create the slant range at the time of the final observation, based on the elapsed time between the observations and assuming a limiting range-rate of  $\pm 80$  km/sec. With the resulting two position vectors, an orbit is created connecting them (using Gauss sector-to-triangle theory for two-body mechanics, or Herget's method for n-body mechanics). The residuals for each observation are calculated, as are the classical orbital elements. The orbit is retained only if the residuals are less than a user-defined threshold, the orbit is prograde (a user-optional constraint), and the orbit is elliptic (a user-optional constraint). Once a successful orbit has been obtained, a new set of noisy observations is created, and the algorithm is repeated until a user-specified number of acceptable orbits have been secured. If 100 variant trajectories are created without satisfying the constraints for a given set of "noisy" observations, the observations are discarded, and a new set is created. If 10,000 variant trajectories (100,000 for two-body mechanics) have been created without satisfying the constraints for any set of "noisy" observations, the algorithm terminates, and the user is advised to increase the RMS tolerance and try again. Typically, the RMS tolerance should be about three times the uncertainty in a typical observation, although lower values are sometimes possible.

Once the desired number of acceptable trajectories has been secured, CODES presents the user with the option of viewing any of five scatterplots:

§ q vs. e

§ q vs. i

§ q vs. w

§ q vs. omega

§ q vs. M

In these plots,

§ variant orbits for which the optical rms residual is between 0 and 1 arc second are **blue**

§ variant orbits for which the optical rms residual is between 1 and 2 arc seconds are **green**

§ variant orbits for which the optical rms residual is greater than 2 arc seconds are **red**

These scatterplots map the orbital solution space, and can help identify multiple solutions; the color of the variant orbits corresponding to each solution can help the user judge which solution is more likely to realistically fit the known observations.

Having generated these variant orbits, the user can also [create ephemeris scatterplots](#), which would allow the user to define the area in which a minor planet might appear at given times; the color of the variant orbits corresponding to each solution can help the user further narrow the likely search region, and hopefully aid in recovery or precovery.

Finally, once these variant trajectories have been created, the user can select [an option in the collision analysis frame](#) which essentially performs a Monte Carlo simulation. Each trajectory is integrated forward to the desired endpoint; any collisions and near-misses are noted. However, there is no trail analysis; and since linearity is not assumed, impact probability is inferred by dividing the number of impacts by the number of trials. While this approach is inelegant, the noteworthy goal is to evaluate the short-term threat of a newly-discovered NEA where only a few observations exist and linearity doesn't apply; such an analysis would likely be impossible otherwise.

Once all of the variant trajectories have been created, the variant with the smallest RMS residual is made the epoch state vector. Note that this will overwrite any pre-existing state vector (the assumption being that no pre-existing definitive state vector is present if we are resorting to SR).

While SR can be used to thoroughly explore the orbital solution space, it should

only be used where linearity does not apply. If linearity *does* apply, Observational Monte Carlo is preferable.

- Compute the observational residuals resulting from the current orbit

As noted above, this option can be useful in [determining the perihelion time offset](#); it can also be used to determine the residuals for a [completely user-specified orbit](#).

The calculations use the same n-body mechanics described above, so it is very important that the user [select the desired number of perturbing bodies](#) from the Level Two Menu before attempting to calculate the residuals. Moreover, if A1 and/or A2 have non-zero values in the current state vector/orbital elements, the contributions from the non-gravitational thrust parameters (including DT) will also be accounted for.

When the residuals calculations are complete, the results are summarized in the file "(minor planet name)\_ compute\_residuals\_output.txt" in the "codes" folder.

The resulting frames successively display the residuals for each observation, the estimated physical properties of the minor planet (deduced from apparent magnitude data, if available), and the current state vector and derived orbital elements. To return to the [Level Two Menu](#) at any point, simply press "Done".

- Propagate the current orbit to a new epoch

This option can be used to propagate the current orbit to another epoch.

As noted previously, the product of initial orbit determination is a state vector (and derived set of orbital elements) referenced to the TDB time of either the first or second selected observation.; this option can be used to propagate the initial orbit to an epoch that is more commonly used or referenced.

If the current orbit includes covariances in the state vector (e.g., a best-fit orbit), these covariances are also propagated to the new epoch.

The calculations use the same n-body mechanics described above, so it is very important that the user [select the desired number of perturbing bodies](#) from the Level Two Menu before attempting to calculate the residuals. Moreover, if A1 and/or A2 have non-zero values in the current state vector/orbital elements, the contributions from the non-gravitational thrust parameters (including DT) will also be accounted for.

When the calculations are complete, the new orbit is saved to disk, and the results are summarized in the file "(minor planet name)\_propagate\_epoch\_output.txt" in the "codes" folder.

The resulting frames successively display the calculated state vector and derived orbital elements. To return to the [Level Two Menu](#) at any point, simply press "Done".

**Important Note:** The set of orbital elements displayed will depend upon whether the user has designated this minor planet as an asteroid or comet. It is therefore vitally important that the user [make the appropriate selection](#) before beginning this option.

- [Display Current Orbit](#)

This option can be used to display (but not to [change](#)) the current state vector (plus covariances) and derived orbital elements (plus uncertainties), referred to the currently specified epoch.

**Important Note:** The set of orbital elements displayed will depend upon whether the user has designated this minor planet as an asteroid or comet. It is therefore vitally important that the user [make the appropriate selection](#) before beginning this option.

To return to the [Level Two Menu](#) at any point, simply press "Done".

- [Manually specify an orbit for this minor planet](#)

In CODES, it is not necessary that the orbit of a minor planet be calculated from observations. This option allows the user to directly enter the state vector (plus covariances) or orbital elements (plus covariances) for the subject minor planet. The user can also specify the epoch, and enter values for the non-gravitational thrust parameters A1 (plus covariance), A2 (plus covariance), and DT.

This user-specified orbit can be used to initialize the [best-fit orbit determination process](#); or it can be used directly to [calculate an ephemeris](#) or to [check for planetary collisions/near-misses](#).

Simply type or paste the desired values into the appropriate data fields. When finished, simply press "Submit", and the new orbit will be saved to disk; CODES will automatically return to the [Level Two Menu](#).

**Important Note:** The set of classical orbital elements available for data entry will depend upon whether the user has designated this minor planet as an asteroid or comet.

It is therefore vitally important that the user [make the appropriate selection](#) before beginning this option.

**IMPORTANT:** *The user should never leave a relevant data-input field blank. If no data is available (e.g., the non-gravitational thrust parameters are not being used), the fields should be filled with zeroes.*

**IMPORTANT:** It is possible to cancel this input function and return to the [Level Two Menu](#); the pre-existing orbit will remain unchanged.

- [Import a minor planet from the MPCORBcr or COMET catalogs](#)

In CODES, it is not necessary that the orbit of a minor planet be calculated from observations. This option allows the user to import the orbital elements from the appropriate MPC database.

This MPC-derived orbit can be used to initialize the [best-fit orbit determination process](#); or it can be used directly to [calculate an ephemeris](#) or to [check for planetary collisions/near-misses](#). Note that, since the MPC databases do not provide values for the non-gravitational thrust parameters, the user would either have to [specify them manually](#), or [solve for them using least-squares](#).

First, the user must be sure to have [designated the subject minor planet as either a comet or asteroid](#). This is vitally important, since CODES will import orbital elements from the MPCOBScr database for asteroids, and from the COMET database for comets.

Next, the user must ensure that the "codes" folder/directory contains a recent version of the Minor Planet Center database of all comets or asteroids for whom reasonably reliable orbits exist:

- For comets, updated versions of the database may be downloaded from <ftp://cfa-ftp.harvard.edu/pub/MPCORB/COMET.DAT> and saved to the "codes" folder/directory as text-file "COMET.DAT".
- For asteroids, updated versions of the database may be downloaded from <ftp://cfa-ftp.harvard.edu/pub/MPCORB/MPCORBcr.ZIP>, unzipped, and saved to the "codes" folder/directory as text-file "MPCORBcr.DAT".

Simply select the minor planet's MPC designation from the drop-down list, and press "Import"; CODES will read the minor planet's orbital elements from the corresponding MPC database, and save them to disk. CODES will then automatically return to the [Level Two Menu](#).

**Important Note:** Due to the size of the MPCOBScr (asteroid) database, the user may

notice a lag associated with selecting and processing this option.

**Important Note:** In some JREs, processing the MPCOBScr (asteroid) database may result in a "java:OutOfMemory" run-time error. This is caused by the JVM heap exceeding its default 64MB maximum size. If this occurs, close the CODES session, and begin a new session with the invocation "java -Xmx600m GUI"; this will increase the maximum size of the JVM heap to 600 MB, which should suffice.

○ Compare observations to positions of known minor planets

This option allows the user to determine if a known comet or asteroid might correspond to the observations of the subject minor planet. The reliability of the results depend upon whether two-body or integrated n-body mechanics are used, how many perturbers are included in the n-body integrations, the level of integration accuracy selected, and whether the epoch of the utilized Minor Planet Center database is near the times of observation. These same factors also impact the time needed for the calculations, as does the far larger size of the asteroid database.

First, the user must be sure to have [designated the subject minor planet as either a comet or asteroid](#). This is vitaly important, since CODES will seek candidates from among completely different databases in each case. Too, if integrated n-body mechanics are to be used, the user should select the desired number of perturbing bodies from the Level Two Menu before attempting this search.

Next, the user must ensure that the "codes" folder/directory contains a recent version of the Minor Planet Center database of all comets or asteroids for whom reasonably reliable orbits exist:

- For comets, updated versions of the database may be downloaded from <ftp://cfa-ftp.harvard.edu/pub/MPCORB/COMET.DAT> and saved to the "codes" folder/directory as text-file "COMET.DAT".
- For asteroids, updated versions of the database may be downloaded from <ftp://cfa-ftp.harvard.edu/pub/MPCORB/MPCORBcr.ZIP>, unzipped, and saved to the "codes" folder/directory as text-file "MPCORBcr.DAT".

For comets, the user can select between two-body and integrated n-body mechanics. The integrated n-body mechanics account for gravitational perturbations (including relativistic effects) from the Sun (including  $J_2$  oblateness), nine planets, Earth's Moon, and [as many asteroids as the user has specified](#); provision is also made for solar radiation pressure. However, since the MPC "COMET.DAT" database does not list the cometary thrust parameters A1, A2, and DT, no use can be made of them here.

For asteroids, the user can also select between using two-body and integrated n-body mechanics. However, since the asteroid database is so much larger, and since



integrated n-body mechanics is so time-consuming, I have made the assumption that the user might be able to use the observed apparent motion of the subject minor planet to classify it as either a NEA, main-belt asteroid, or Centaur/TNO. Thus, I have given the user the option to narrow the candidate search accordingly (thus reducing processing time) when using integrated n-body mechanics. Additionally, I've provided options to narrow the n-body candidate search based on estimates of the minor planet's absolute magnitude, which can be determined as part of the best-fit orbit determination process.

For both comets and asteroids, the user can substantially reduce n-body processing time by selecting "medium" integration accuracy. Whereas the "high" accuracy option ensures that the local error in final position will not exceed 0.9 km, the medium option restricts the local error in final position to 0.1 Earth radii; however, this larger position error (usually smaller than ignoring the parallax effect on an Earth-bound observer relative to geocentric) only translates to 0.06 arcseconds at 1.5 AUs, and to 33 arcseconds at 1 lunar distance. Except for very fast-moving (and thus presumably very close) NEOs, this tradeoff is usually quite feasible, especially given the substantial reduction in processing time. If necessary, any candidates identified in a medium accuracy n-body search can be reevaluated at high accuracy by creating a new MPCOBScr.DAT or COMET.DAT file in which all non-candidates are removed.

Once the user clicks "Process", CODES reads each entry in the appropriate database, extracts the orbital elements, uses either two-body or n-body mechanics to predict the astrometric position of the catalog object at the TDB time of each observation of the subject minor planet, and calculates the RMS residual. If the RMS residual is less than three arc degrees, and if the residual for each observation is also less than three degrees, the catalog object is classified as a candidate. An ordered list of all candidates is compiled, and the full candidate list is reported to the user upon completion in both screen and file (MPC\_compare\_output.txt in the "codes" folder/directory) output.

When finished reviewing the screen output, simply press "Done"; CODES will return to the [Level Two Menu](#).

- Generate an optical ephemeris

This option allows the user to either create an astrometric ephemeris for the subject minor planet, or create RA vs. Dec scatterplots to aid in precovery/recovery searches.

To create a topocentric ephemeris, the user must specify an observatory code or the observer's position (latitude, longitude, altitude); a geocentric option is also available. The user must also specify the UTC start and end dates, as well as the interval at which ephemerides should be generated.

In generating the predicted ephemerides, CODES uses the last saved state vector for the subject minor planet:



- Integrated n-body mechanics is used, accounting for gravitational perturbations (including relativistic effects) from the Sun, nine planets, Earth's Moon, and as many asteroids as the user has specified; as a result, it is very important that the user [select the desired number of perturbing bodies](#) from the Level Two Menu before attempting to calculate the ephemeris. Provision is also made for solar radiation pressure. Additionally, if the last saved state vector has non-zero cometary thrust parameters (A1, A2, DT), their influence will also be accounted for.
- Finally, if the visual absolute magnitude and slope parameter are non-zero, CODES will estimate (for phase angles between zero and 120 degrees) the predicted apparent visual magnitude for each entry. However, since CODES uses different algorithms in calculating the visual apparent magnitude for comets and asteroids, it is very important that the user [designate the subject minor planet accordingly](#) from the Level Two Menu before attempting to calculate the ephemeris.

The following data are generated for each entry in the ephemeris:

- Astrometric Right Ascension and Declination
- Predicted Visual Magnitude
- The position angle and the semi-major and semi-minor axes of the 3-sigma error ellipse (assuming covariance data exists for the orbital elements/state vector of the subject minor planet)
- The speed and position angle of the minor planet's apparent motion
- The observer-minor planet distance(delta), the Sun-minor planet distance (r), and the apparent elongation of the minor planet from the Sun as seen by the observer.

When calculations are complete, CODES displays the ephemeris in tabular format. A file version of the ephemeris is also created at "ephemeris\_output.txt" in the "codes" folder.

Alternatively, if the user has previously generated a sampling of valid orbits using [Observational Monte Carlo](#) or [Statistical Ranging](#), CODES can generate a series of RA vs. Dec scatterplots for each ephemeris entry. Each scatterplot is centered on the nominal predicted position, with the color-coding of each "virtual orbit" reflecting its RMS position error, [as described above](#). The distribution of "virtual orbits", as well as their color, can indicate the likely search area for the precovery or recovery of a minor planet on imagery.

**Note:** CODES supports observations and predictions for the range 1800-2200.

The user may click "Done" to return to the [Level Two Menu](#).

- Check for collisions and/or near-misses

Once all [available observations have been processed](#), and either [a best-fit orbit has been calculated](#) or a [sampling of variant orbits](#) has been obtained, this option allows the user to identify potential collisions and/or near misses that the subject minor planet may experience with the Sun, nine planets, or Earth's Moon. Two search methods are available: The linear method is limited to encounters experienced by the current nominal state vector, while the nonlinear method is a more exhaustive survey employing a user-specified number of variant state vectors. The user should select the radio button next to the desired method.

In each case, the user must specify the threshold (in A.U.s) for a near-miss, as well as the desired integration accuracy. The search can be limited to events involving Earth, if desired (especially useful in limiting processing time). When the user clicks "Select" next to the desired search timeframe, CODES begins searching for collisions and/or near-misses from the epoch of the last-saved state vector to the desired endpoint.

In the linear method, CODES begins propagating the nominal state vector and covariance matrix from the current epoch to the desired endpoint. Integrated n-body mechanics is used, accounting for gravitational perturbations (including relativistic effects) from the Sun, nine planets, Earth's Moon, and as many asteroids as the user has specified; as a result, it is very important that the user [select the desired number of perturbing bodies](#) from the Level Two Menu before attempting to calculate the ephemeris. Provision is also made for solar radiation pressure. Additionally, if the last saved state vector has non-zero cometary thrust parameters (A1, A2, DT), their influence will also be accounted for.

At the end of each integration time-step, CODES checks whether the distance between the subject minor planet and a solar system body (Sun, nine planets, Earth's Moon) is less than the radius of that solar system body; if so, CODES assumes a collision has occurred, uses bisection to determine the precise moment of collision, and uses the state vector covariances to estimate the probability of collision. If no collision is detected at the end of the integration time-step, CODES looks for sign changes in the first derivative of the distance to each solar system body that might indicate a relative minimum in distance; if found, bisection is used to determine the precise minimum distance, and this minimum distance is compared to the radius of the solar system body to determine whether a collision (distance < radius) or near-miss (distance < near-miss threshold) has occurred. Again, if a collision is predicted, CODES uses bisection to determine the precise moment of collision, and uses the state vector covariances to estimate the probability of collision. If a near-miss is predicted, CODES again uses the state vector covariances to estimate the probability of collision.

**Important:** Multiple close encounters can significantly magnify the growth of state vector covariances with time, and substantially reduce the reliability and relevance of long-term forecasts. Indeed, once state vector covariances grow to 0.05 AUs or more, predictions as to collisions/near-misses are of dubious value. Ultra-precise radar

observations can mitigate this problem somewhat. Nevertheless, for recently-discovered minor planets with relatively large state vector covariances, users may want to limit the search for collisions/near-misses to relatively short timeframes, until additional observations help refine the trajectory and make distant forecasts more meaningful.

**Note:** Linear collision analysis does not require the prior processing of all available observations; in fact, so long as the user has [entered](#) or [imported](#) a state vector or set of orbital elements (plus covariances), linear collision analysis can be performed.

In two of the nonlinear methods, CODES creates a user-specified number of variant state vectors; depending on user selection, these variant state vectors are either distributed normally (in observation space) about the nominal epoch state vector, or distributed uniformly along the Line-of-Variations (roughly corresponding to the major axis of the epoch state vector uncertainty ellipse). Each of these variant state vectors is propagated forward from the current epoch to the desired endpoint, and its collisions and/or near-misses are noted precisely as in the first method. These events are compiled and sorted into dynamically-distinct trails by target body, date, semi-major axis, and miss distance. If a reversal is noted (in which successive VAs march toward a target, then reverse direction and begin to recede), a dense sampling is performed around the point of reversal to ensure that the point of closest approach (or point of collision) has been determined. Otherwise, differential correction is applied to the nearest VA of each trail to iteratively reduce the miss distance; if the variant state vector can be adjusted (along the LOV) so as to cause a collision, it is termed a Virtual Impactor (VI), and linear techniques are used to estimate the probability of collision. Even when the differential correction process does not force a collision, we will very often identify near-miss encounters unique to the particular variant trajectories. In short, by thoroughly sampling the uncertainty region surrounding the nominal epoch state vector, the user can increase the degree of confidence that all statistically-significant collision/near-miss events have been identified.

In the final nonlinear method, a sampling of variant trajectories previously obtained using the Statistical Ranging (SR) algorithm is used in a Monte Carlo simulation. Each trajectory is integrated forward to the desired endpoint; all collisions and near-misses are noted, but no trail analysis is performed. Since the geometry is inherently non-linear, the impact probability is inferred simply by dividing the number of impacts by the number of trials. The intent here is to evaluate the short-term threat posed by a newly-discovered NEA where only a few observations exist and where linearity doesn't apply; once the observational arc has been extended and a least-squares best-fit orbit has been obtained, the other nonlinear collision algorithms can more fully characterize the medium and long-term threat..

**Note:** If the Statistical Ranging option is used, CODES will ignore the contents of the text-input area on this screen specifying the number of variant trajectories to be created; the SR trajectories have [already been generated](#), and only these sampled trajectories will be used.

Note that linear and non-linear collision analysis should be complimentary. If no intervening perturbing close approach exists, and if the covariances at the time of the event are sufficiently small, linear analysis may well characterize the event sufficiently. But if a particularly close planetary encounter disperses the variant trajectories, subsequent events can become chaotic and dependent on the exact path taken, making nonlinear collision analysis necessary.

**Important:** Whether the variant state vectors are distributed normally or along the Line of Variations, least squares techniques are employed to ensure that the resulting variant state vector has the smallest possible residual. As a result, it is **vitaly important** that the user [input all available optical and radar observations](#) **prior** to attempting nonlinear collision analysis. Simply [specifying](#) or [importing](#) a nominal state vector is not sufficient; CODES must have processed and stored the actual observations before nonlinear collision analysis is possible. Ideally, the user will have also used CODES to calculate the [initial](#) and [best-fit](#) state vectors; since the observations are then consistent with the nominal state vector, the least-squares-driven collision analysis algorithm should be quite stable.

**Important:** The use of multiple variant state vectors and iterative differential correction results in a great deal of numerical integration; as a result, the user should be aware that [this method requires significant processing capacity](#). Since the variety of user platforms makes it difficult to offer universal guidance, I would suggest that users run a benchmark test using 100 variant state vectors over a 25 year timeframe; knowing the time required to process this small survey, users should then be able to judge the realistic number of variant state vectors that their platform can support over the full range of survey timeframes.

Processing time can be reduced by utilizing the "target filtering" option to filter out events involving other planets, and consider only Earth-related events. It should be noted, however, that this should probably not be a default selection, since comet Shoemaker-Levy 9 amply demonstrated that Earth is not the only planet subject to bombardment. Indeed, Earth-related events very often have counterparts involving the Moon; and the relatively small separation involved makes these events particularly interesting.

The processing demands can also be somewhat mitigated by reducing the integration accuracy. "High" accuracy ensures that the local error in the final position will not exceed 0.9 km; however, in determining whether a collision has occurred with a planet 6378.14 km in radius, this level of accuracy is not always necessary. "Medium" accuracy ensures that the local error in the final position will not exceed 0.1 Earth radius; this option can reduce processing time by a factor of 5 (or more) over a 50-year search period, and should capture most of the same events as the "high" option (though with less precision). My recommendation would be to search initially with the "medium" option; if any Virtual Impactors or particularly close near-misses are observed, the search can be repeated at "high" accuracy.

**Very Important:** Predictions using "medium" accuracy should not be relied upon in

the case of a NEO such as 2000 SG344, which has multiple successive near-miss encounters over the next 100 years. Unless the cumulative perturbations of these encounters are accurately modeled with high integration accuracy, predictions are of dubious value.

I have spent a great deal of time testing the non-linear collision analysis algorithms, and I believe CODES' predictions are now comparable to those of Sentry or NEODyS. Note that I said "comparable". There are at least four reasons why our predictions could differ:

- The systems begin with slightly different state vectors;
- CODES collision analysis occurs in (and results are quoted in) 3-space, while Sentry and NEODyS collision analysis is performed in the impact plane;
- The differential correction and dense sampling algorithms employed by CODES differ from those outlined by Andrea Milani in his series of papers on Virtual Impactors.
- CODES assumes that the uncertainties in optical ra/decs are 1.0 arc seconds by default, while Sentry often uses smaller values; as a result, the width of the event covariance regions and the impact probabilities may differ substantially.

Far from rationalizing these differences, I believe it is extremely useful to independently check the predictions made by each system.

In evaluating high-risk asteroids, I would suggest the following settings:

**- For Line-of-Variations analyses**

Nonlinear Collision analysis - Consider 1201 (or more) variant trajectories  
Sample Line-of-Variation  
Set near-miss threshold 0.05 AUs  
High integration accuracy

**- For Monte Carlo analyses:**

Nonlinear Collision analysis - Consider 1201 (or more) variant trajectories  
Sample normal distribution  
Set near-miss threshold 0.05 AUs  
High integration accuracy

Because of processing limitations and the inevitable growth of covariances over time, I would suggest that the timeframe generally not exceed 100 years. As more distant future events are greatly affected by even very small changes in the epoch state vector, I would suggest that only asteroids observed by radar (and thus with very small state vector covariances) should be considered for analyses beyond 2100.

When calculations are complete, CODES displays the predicted collision/near-miss events in tabular form, with UTC date, nominal and minimum distance of closest approach, 3-sigma width of the event covariance ellipse, probability of collision, and the largest multiple of sigma for the components of the (variant) state vector. A file version of the output is created at "[minor planet]\_collision\_output.txt" in the "codes" folder; this file also provides the epoch state vector of each VI and near-miss

trajectory.

**Note:** CODES supports observations and predictions for the range 1800-2200.

The user may click "Done" to return to the [Level Two Menu](#).

- Change the number of perturbing bodies for integrator

This option allows the user to specify the number of perturbing bodies to be used in integrated n-body mechanics. Since integration is used extensively in CODES (e.g. [determination of best-fit orbit](#), [checking for collisions/near-misses](#)), this selection should be made prior to attempting any operation.

In most cases (e.g., orbit determination using optical observations, especially over only a short time arc), the (default) second selection is acceptable. This provides high-accuracy perturbations due to the Sun, nine planets, Earth's Moon, and the three largest asteroids (Ceres, Pallas, and Vesta).

If very high accuracy is required (e.g., careful analysis of a possible planetary impact, availability of optical observations over a long time arc, or availability of radar observations), I would recommend using the third selection (all asteroids with reliable mass estimates). Note that this will greatly increase the number of calculations per integration step, so **expect much slower processing with this setting.**

Experience has shown that, when analyzing the multiple close encounters (i.e. near-misses and collisions) of a minor planet with major solar system bodies, the accuracy of predicted events depends more on the accuracy of the epoch state vector than on the ultra-precise modeling of the later perturbations. Thus, in high-accuracy cases, I would emphasize using the third selection (all asteroids with reliable mass estimates) in the determination of the minor planet's best-fit orbit.

**Note:** The positions and velocities of the Sun, nine planets, and Earth's Moon are calculated using the JPL DE405 ephemeris.

- Specify the radius/visual brightness parameters

The radius, visual absolute magnitude, and slope parameter are used by CODES in several operations, such as [creating an ephemeris](#) or processing radar observations.

When the [best-fit orbit](#) of the subject minor planet is calculated, CODES uses visual apparent magnitude data (if available) to estimate (using different algorithms, depending on whether the object is a comet or asteroid) the visual absolute magnitude and slope parameter. For asteroids, the slope parameter is used to estimate the albedo, itself resulting in an order-of-magnitude estimate of the radius.

However, if the user has instead [imported](#) or [manually entered the last-saved state vector/orbital elements](#), then the radius, visual absolute magnitude and slope parameter are unknown. This option can thus be used to specify these parameters. Additionally, if a [best-fit orbit](#) has already been calculated, the user can specify only the slope parameter, and ask CODES to calculate the resulting visual absolute magnitude and (for asteroids only) radius; since different algorithms are used for comets and asteroids, however, it is very important that the user [designate the subject minor planet as a comet or asteroid](#) before attempting this calculation.

The default radius is 1 km. However, using this value for a large comet, for instance, could seriously degrade the results of radar calculations. For comets, use the best available estimate of the radius of the coma (from which radar signals can be expected to deflect).

When the correct values have been entered into the data fields, simply click "Set". CODES will save the data to disk, and return to the [Level Two Menu](#).

## Where to find data for use in CODES

You can find actual observations of minor planets in the Minor Planet Electronic Circulars (MPECs) published by the Minor Planet Center at Harvard Smithsonian. The URL is: <http://cfa-www.harvard.edu/mpec/RecentMPECs.html>

The MPC also provides an Extended Computer Service called MPCOBS which enables users to extract files containing all observations of a specific minor planet. Application for access can be made from their web site at <http://cfa-www.harvard.edu/iau/services/ECS.html>.

Another excellent source is the Near-Earth Objects-Dynamic Site (NEODyS) at: <http://newton.dm.unipi.it/cgi-bin/neodys/neoibo>

Astrometric radar observations are still fairly rare, but they are increasing in both frequency and importance. You can find all astrometric minor planet radar observations made up to the present time at URL: [http://ssd.jpl.nasa.gov/radar\\_data.html](http://ssd.jpl.nasa.gov/radar_data.html)

You can also generate synthetic observations using JPL's HORIZON ephemeris system at URL: <http://ssd.jpl.nasa.gov/cgi-bin/eph>

## Notes on numerical integration and the accuracy of CODES

- Numerical Integration

I gave great thought to the selection of a numerical integration scheme. Variable step sizes and local



error control are available for both single-step and multi-step methods.

Many applications (including the JPL DE405/406 ephemeris) use multi-step methods, since only a single functional evaluation is needed per step; the disadvantage of such methods is that the permissible step sizes are often smaller, most especially at start-up.

For this particular application, I wanted an integrator that would cover up to 200 years of collision analysis as quickly as possible; in other words, I needed an integrator optimized for minimum processing time over long spans of integration. In my trade studies of Embedded Runge-Kutta (ERK) single step methods vs. Adams predictor/corrector multistep methods, the Dormand-Prince 8(7) ERK method emerged as the winner.

I'd like to thank Bill Gray for sharing the results of his similar comparisons, and for suggesting a "tabular integrator" that may be implemented in the future.

- Accuracy of CODES

My intent in developing this software was to make few, if any, compromises in accuracy. At the moment, there are seven approximations used:

- In calculating the positions and velocities of the perturbing asteroids, I only use a single correction for light time-of-flight, rather than an iterative correction. Since the positions and velocities are based on the mean orbital elements, I don't believe the accuracy (and CPU cycles) of an iterative loop is required.
- In evaluating the differential equation of motion for the subject minor planet, I exclude the  $1/c^2$  terms involving the Newtonian acceleration of each perturbing planet and asteroid, due to the gravity of the other bodies in the solar system. Tests over a 200 year span showed absolutely no contribution due to this term at the 0.01 arc second level; thus, there should be no effective impact on accuracy. Moreover, calculating the necessary accelerations represented over 80% of total CPU time.
- In evaluating the differential equation of motion for the subject minor planet, I account for the Sun's  $J_2$  (oblateness) perturbations, as well as the acceleration caused by the impact (and subsequent reflective recoil) of solar radiation, including the relativistic Poynting-Robertson effect. However, I do not account for non-radial accelerations caused by later thermal emissions (the Yarkovsky effect); since modeling this phenomenon requires knowledge of the minor planet's radius, density, rotational period/axis, and surface/subsurface thermal properties, I regard this as a possible later enhancement.
- In estimating the radius of an asteroid (computed along with the visual absolute magnitude and slope parameter), the presumed albedo is estimated as a function of the slope parameter. The intent here was to squeeze an order-of-magnitude estimate of the radius from the optical astrometric and visual magnitude observations. Obviously, albedo can only be definitively measured if the radius is already known, although a much better estimate of the albedo can be made if spectroscopic observations determine the asteroid class. In the future, I may add the ability for the user to specify the albedo if it is independently available, and then calculate a better estimate of the radius. For now, though, the user should understand that the estimated radius is an order-of-magnitude approximation only.



- In CODES, calculations are made in the ICRF/J2000 reference system, which became the IAU standard in 1998. Previously, optical observations had been referred to the Fundamental Katalog 5 (FK5) and the J2000 equinox. CODES treats the ICRF and FK5 reference frames as being equivalent. In fact, the difference between them is less than 0.01 arc second; but since this is an order of magnitude smaller than the precision of most current optical astrometric data (approx. 0.1 arc second), this seems a safe approximation. And since optical astrometry now uses ICRF/J2000, future advances in precision will not require the addition of an FK5->ICRF transformation.

## Data Input Errors

It was decidedly not my intent to design a fool-proof user-interface. Rather, the intent was to create a fast and efficient data-entry and analysis tool for an orbital analyst.

Thus, if absurd data (such as specifying that a minor planet has an orbit with  $a = 0$  and  $e = 0$ ) is input, or if a data field is left blank, the software may quite possibly hang. However, recovery is sometimes as simple as entering valid data and resubmitting; in the worst case, CODES must be closed and restarted, with all completed data entry prior to that point having been saved to disk.

**IMPORTANT:** *The user should never leave a relevant data-input field blank. If no data is available (e.g., the user does not have Earth Orientation Parameter data for an observation), the fields should be filled with zeroes.*

## Release Information

This is CODES version 4.2, incorporating all planned baseline features (including non-linear collision analysis), plus improved algorithms for short-arc orbit determination, and the BC-405 integrated asteroid ephemeris.

Planned upgrades include the following:

- Extending the era in which observations are supported back to 1700
- Improved MPCORBcr search capability to accommodate observational epochs differing substantially from the catalog epoch
- User entry of barycentric equatorial observer coordinates to accommodate SOHO observations
- Determination of impact lat/long in collision analyses
- Processing of Astrometrica observation files
- Additional asteroid perturbors

- Implementation of JPL DE406 planetary ephemeris
- Collision analysis through 3000 A.D.
- Implementation of the Yarkovsky effect

Additional suggestions would be greatly appreciated.