Detailed User guide for the Simple MCMC Orbit Determination Tool (SMODT)

Kyle Mede

March 31, 2015

Contents

1. Introduction

These are the outlines of the functions that act as the models to calculate the orbital elements for either a binary star system or planet orbit. The equations are the result of combining Kepler's laws, the definitions of the orbital elements and the naturally occurring symmetries and rules of a stable binary system.

There are both Python and C++ versions of these functions/models. The bulk of the multi-process and file management, post analysis and plotting of the results is done in Python while the computational stages of the simulation are done in C++ to take advantage of its speed. A more detailed description of these issues and the simulator will be in another document to be written later. All three astrometry models described here have been tested and produce the same resulting values/fit to within approximately 10 significant figures, well past the accuracy limits placed on the parameters by their associated 68% or 95% errors.

1.1. Measured Astrometric Values

There are multiple ways to visually represent an astrometric system, and thus the measured values must be clearly understood to ensure they are correct before using them as inputs to the SMODT orbital model and plotting routines.

The orientation and definition of x and y depends on the choice of view of the binary system. The two most common of these are the telescope and the naked eye view, seen in Figure 1 compared to a standard Cartesian coordinate system. As direct observations of binary systems result in images matching the orientation of the telescope's view, and that in many different situations angles are commonly measured from the positive "x" axis, the combination of these two led to the standard conventional coordinate system given by a) in Figure 2; thus, E=y and N=x. This convention can also be seen used in Figure 3, showing the geometric meaning of the Thiele-Innes elements along with the apparent and true orbital ellipses of an example binary system. While that is the conventional coordinate system for measuring astrometric data, when plotting the data along with the predicted orbit more recently people follow that of b) in Figure 2. Therefore, as long as the proper coordinate systems are being used the appropriate x and y values can be calculated with the equations (33a & 33b).

To help avoid the possible confusion due to the different definitions of x and y, the more easily understood ϕ and ρ, with associated errors, are used as the input data coordinates. The models and post-processing plotting routines will then appropriately handle the required conversions.

In the case where the data is measured as the difference in Right Ascension (α) and Declination (δ) of the companion from the primary star, in units of [$''$], these match the E=y and N=x respectively. This allows for direct comparison of the data to the values calculated in equations (31a) and (31b). If these units need to be converted to ϕ and ρ, then the following conversions can be used.

The respective x and y values from the measured position angle (ϕ) and separation angle (ρ) are given by:

$$x_{TH-I} = \rho \cos(\phi) = y_{plot} = North = \delta \tag{1a}$$

$$y_{TH-I} = \rho \sin(\phi) = x_{plot} = East = \alpha \tag{1b}$$

with ϕ in units of [radians] and ρ in [$''$].

The error in the data values in the new coordinate system would then be:

$$\sigma_x = \Delta x_{TH-I} = x_{TH-I}\sqrt{\left(\frac{\Delta\rho}{\rho}\right)^2 + \left(\frac{\cos(\phi+\Delta\phi)-\cos(\phi)}{\cos(\phi)}\right)^2} \tag{2a}$$

$$\sigma_y = \Delta y_{TH-I} = y_{TH-I}\sqrt{\left(\frac{\Delta\rho}{\rho}\right)^2 + \left(\frac{\sin(\phi+\Delta\phi)-\sin(\phi)}{\sin(\phi)}\right)^2} \tag{2b}$$

If the astrometric values measured are as East and North, both in units of[$''$], the following equations can be used to convert them to those of ϕ and ρ:

$$\rho = \sqrt{(East)^2 + (North)^2} \tag{3a}$$

$$\phi = arctan\left(\frac{East}{North}\right) \tag{3b}$$

with matching errors:

$$\sigma_\rho = \Delta\rho = \rho\left(\frac{|\Delta North \times North| + |\Delta East \times East|}{North^2 + East^2}\right) \tag{4a}$$

$$\sigma_\phi = \Delta\phi = \left|\frac{East}{North}\right| \frac{\sqrt{\left(\frac{\Delta East}{East}\right)^2 + \left(\frac{\Delta North}{North}\right)^2}}{1 + \left(\frac{East}{North}\right)} \tag{4b}$$

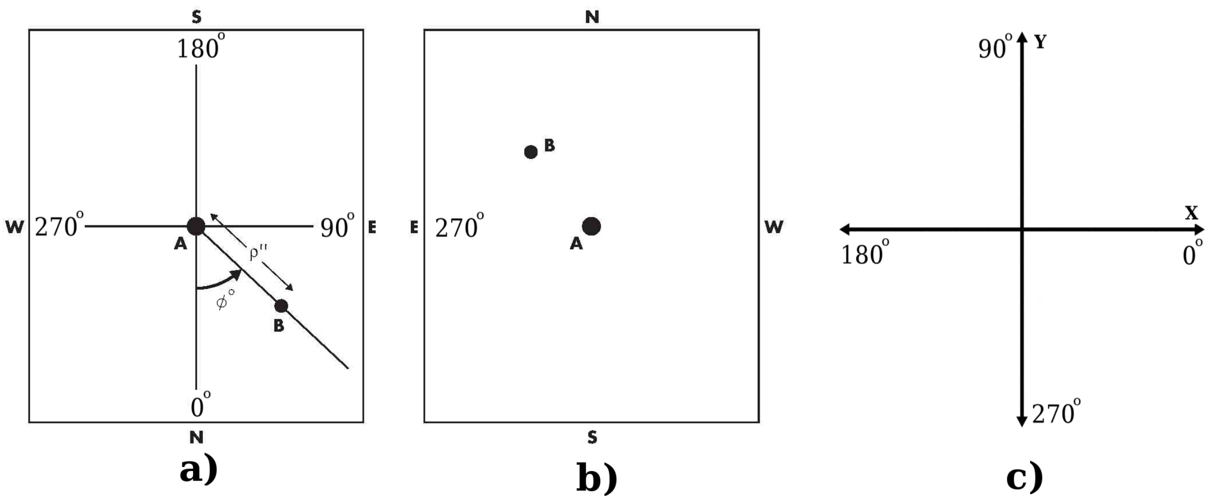

Fig. 1.— View of a binary system through a) A telescope, b) The naked eye, compared to c) The Cartesian coordinate system. Argyle (2004).

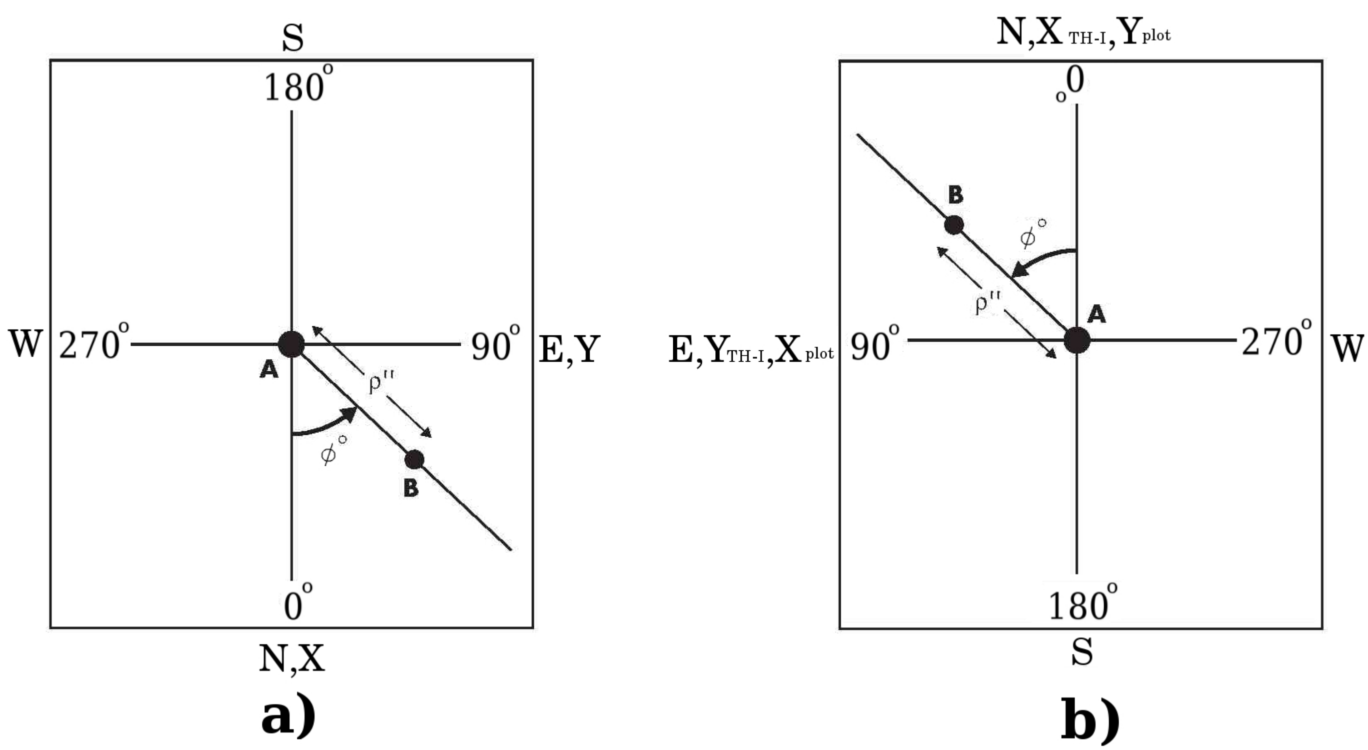

Fig. 2.— Conventional orientations and coordinates for a) measuring and b) plotting the astrometric values of a binary system.

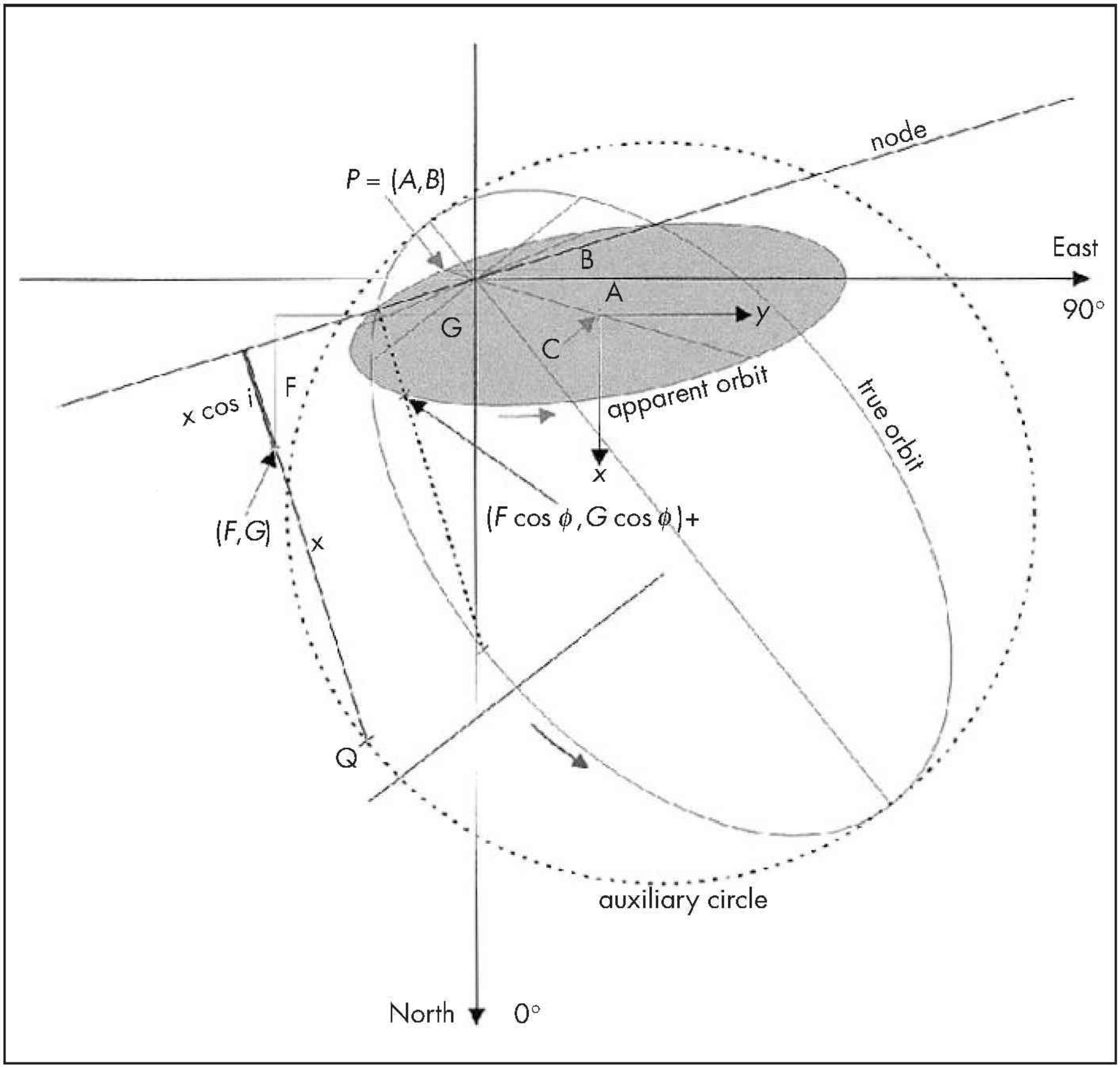

Fig. 3.— Plot of apparent and true orbital ellipses, and the Thiele-Innes elements, Argyle (2004).

1.2. Binary Orbital Elements

When observing a binary system with either the naked eye, or with a telescope, what can be seen is their positions in the plane of the sky, see Figure 1. As was discovered by Johannes Kepler in the early 17^{th} century binary systems will move in elliptical orbits around the center of gravity. In most cases though, the elliptical path seen, commonly referred to as the ‘apparent ellipse’, will be that orbit projected onto the plane of the sky. This is because most orbital systems do not orbit perfectly in the plane of the sky and instead lie in the system’s ‘orbital plane’. So, with a sufficient number of observations over a long enough amount of time, the apparent ellipse can be drawn, but not the true orbital path in the orbital plane. In order to find the exact shape of the true ellipse and its orientation with respect to the plane of the sky, the 6 parameters called the ‘orbital elements’ must be calculated.

The orbital elements that describe the orientation of the orbital plane to the plane of the sky are the inclination, i, and the longitude of the ascending node, Ω. As can be seen in Figure 4, i is the angle between the the two planes centered on the intersection line. In the case where the position angles measured are increasing with time, the orbital motion is considered direct, or prograde, and i will lie in the range 0°-90°. On the other hand, if they decrease with time, the motion is retrograde and it will take values of 90°-180°, (Heintz 1978). The intersecting line between the two planes is known as the ‘line of nodes’ with the ascending node being the point on the orbit passes upward through the plane of the sky, and the descending node being on the other side when the orbit passes through moving downwards. Most commonly measured in the plane of the sky from North to East, Ω is the angle to the ascending node, shown as ‘N’ in Figure 5; due to this, Ω is sometimes referred to as the position angle of the line of nodes, (Binnendijk 1960). Unfortunately, just by looking at the system there is no way to tell if the body is truly moving upward at a the ascending node due to the apparent orbit being a projection onto the sky. To handle the problem temporarily, a convention of Ω being between 0°-180° is commonly used. This 180° discrepancy can be removed when radial velocity data also exists for the system. If the orbital motion of the object is directed away from the Earth, shown by positive velocities, that node is confirmed to be the ascending node and the convention holds. Else, the node is actually the descending node and the location of the ascending node would be within 180°-360°, (Heintz 1978).

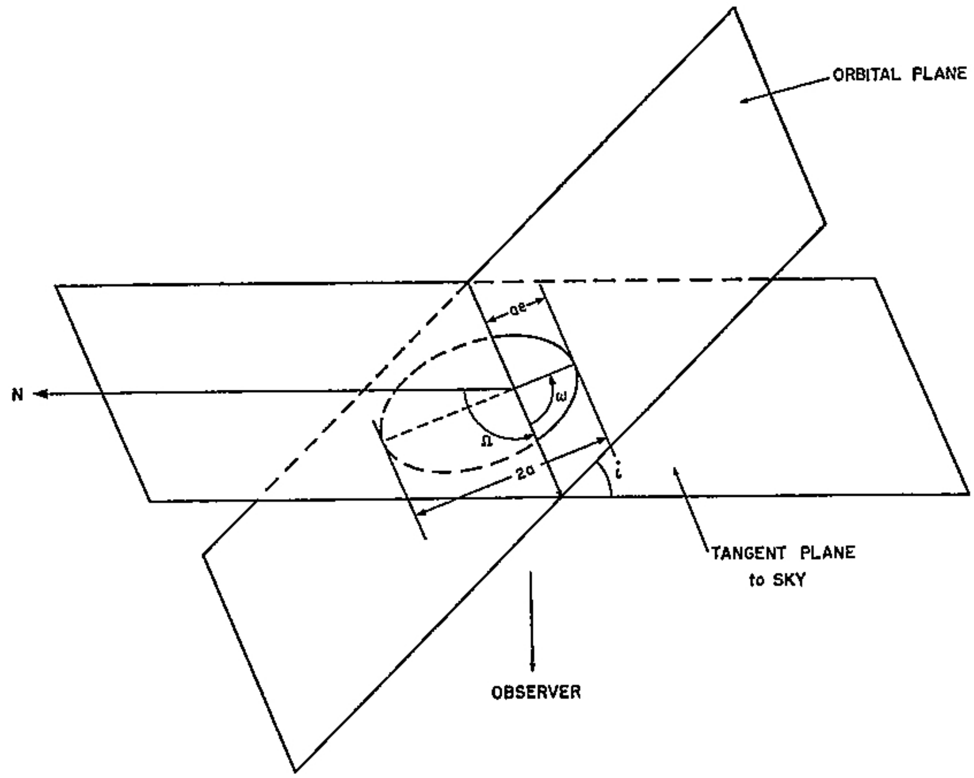

Fig. 4.— Orbital plane and the tangent plane to the sky From Batten (1973).

The two orbital elements that define the shape of the true orbital ellipse are the eccentricity, e, and the semi-major axis, a. Looking at Figure 6, the semi-major axis half the length of the line between the two most distant points on an ellipse. The two focus points along this line are represented by F and F' with the distance from the center of the ellipse to either of these being simply ae. The value of the eccentricity for an ellipse take values of [0,1.0], with the value 0 making it a circle and 1.0 a parabola. For a binary system, both of the bodies are actually moving with respect to a common center of gravity, see Figure 7. A standard convention to simplify this is to put the larger body at a focal point and consider the apparent path of the other body as the combined elliptical orbit. By doing the math, one can check that this is a valid convention choice assuming a=$a1$+$a2$.

The last two orbital elements needed to fully describe an orbit in three dimensional space are the mean anomaly, M, and the argument of periapsis, ω. The point of periapsis is where the two bodies are the closest to each other, shown as Π in Figures 5 and 6. ω is then just the angle between the ascending node and Π measured in the orbital plane, (Heintz 1978); sometimes this angle is referred to as the longitude of periapsis to match the naming of the longitude of the ascending node, (Hilditch 2001). The mean anomaly, M defines the position of the secondary body through its orbit,

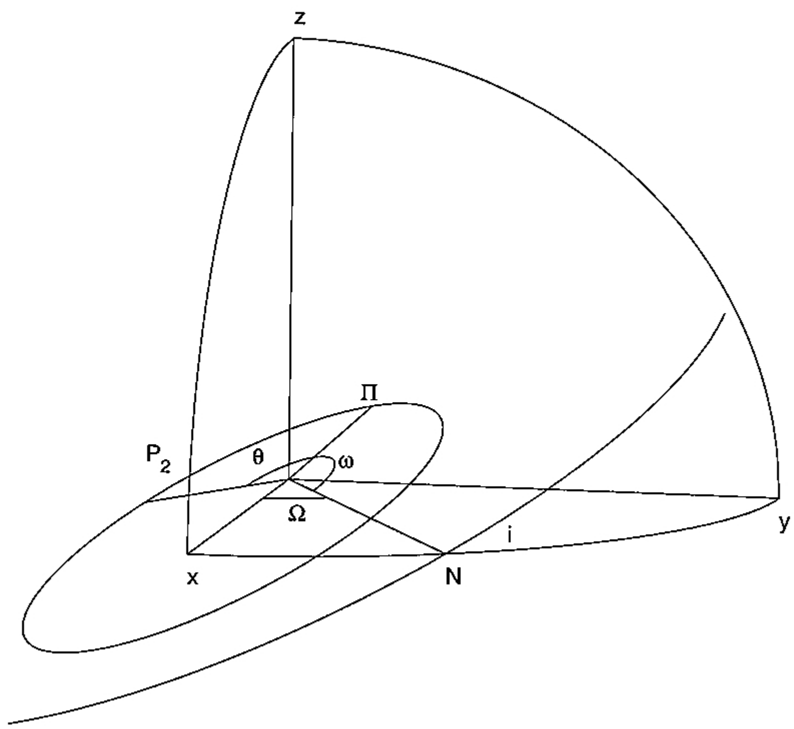

Fig. 5.— The relative orbit of a binary located in three dimensions to show the orbital elements of its orbit. From Hilditch (2001).

but not as a directly measurable value, other parameters are commonly chosen to replace it in the orbital elements. M is calculated following equations (5a & 5b), where n is the mean motion, t is the time the system is being observed and T is the time the second body was at the periapsis point, usually called the time of last periapsis. The other parameters more commonly used are the period of the orbit and T. By using these parameters the semi-major axis is not strictly required if the mass of the two bodies is known, as Kepler's third law of motion can be used to calculated it, shown in Equation (6), where M_1 and M_2 are the masses of the two bodies.

$$n = \frac{2\pi}{P} \tag{5a}$$

$$M = n\left(\frac{(t-T)}{365.25}\right) \tag{5b}$$

$$P^2 = \left[\frac{4\pi^2 a^3}{G(M_1 + M_2)}\right] \tag{6}$$

Therefore, either [i, Ω, e, a, ω, M] or [i, Ω, e, ω, P, T] can be used to fully describe any binary

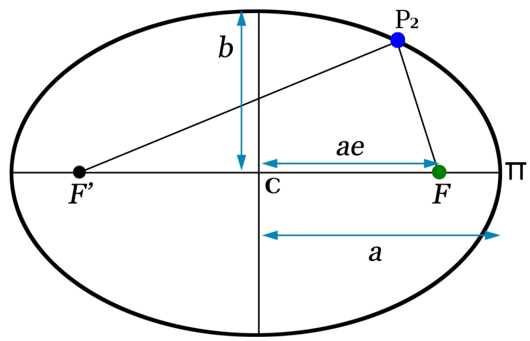

Fig. 6.— The ellipse of an orbit in the orbital plane.

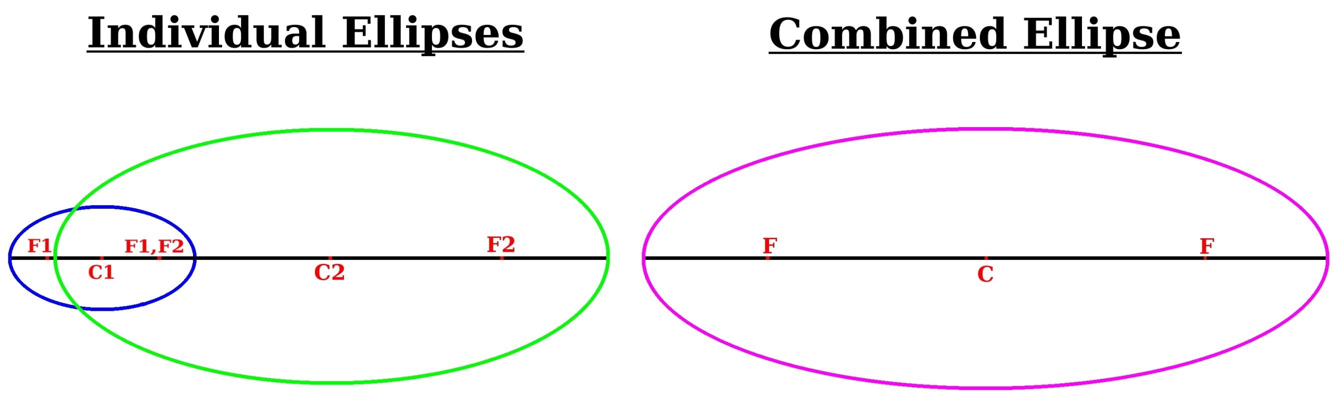

Fig. 7.— A comparison of a single combined ellipse to the original individual ellipses for a binary system.

orbit in space. Later in Section 4 some of the orbital models for converting between the observed apparent ellipse and the true ellipse will be described.

2. Simulator Settings

Here we will go over the large array of settings in the settings file and the possible modes to run the simulator in.

First, lets go through ALL the settings in the "SimSettings" file, as it is where the user is expected to spend the bulk of their time playing around to get things to run the way they want.

chiSquaredMax
This will set the maximum allowed reduced χ^2 allowed during simple Monte Carlo runs. During Simulated Annealing it also poses a use to allow the starting parameters to jump to a new set if the chain has yet to find a solution under that value, in this case trial and error are needed but a value around 300 might work at the start and reduced as the parameters are constrained. That situation can occur if you have a low starting temperature, or the initial parameters are unfortunately in a very bad section of the parameter space. Thus, the jumping routine helps get out of those ruts and this parameter determines if it is in said rut.

numSamples
The total number of samples to produce for a single chain. This is the number of samples for MCMC, Simulated Annealing or Monte Carlo if running in those modes, while the separate Simulated Annealing samples can be controlled with the "numSamples_SimAnneal" when running in MCMC mode. Remember that if you use multi-processing, then the true total number of samples will be this number multiplied by how many processes/chains you run.

numSamplePrints
The number of times a print block will be displayed to the screen of the vital update information for how the simulation is running. These will occur at sample = numSamples/numSamplePrints intervals during the simulation.

useMultiProcessing
Turn on multi-processing? If set to true, then there will be multiple chains started. The current code will start # chains = total number of virtual cores available - 1, leaving one core free to make sure the computer OS still runs smooth. ie. on an computer with a 8 virtual cores (4 cores with 2 threads each), 7 cores will run a chain at 100% capacity and 1 will be left remaining to do anything else. If the user wishes, they can go into the code and change the single line that controls the number of cores used to suite their needs. Please note that each core will be running at ~100%, and thus create a lot of heat. So, make sure you have a sufficiently capable cooler to avoid CPU failure.

silent
There are two levels of extra information printing above that of the print block controlled with "numSamplePrints", or any error prints due to problems. The higher level one is "silent", with

some extra prints that will occur for each sample and is useful for some debugging.

verbose
This is the second level of debug print messages. It will basically trigger the simulator to give you a high level of verbosity to its step by step process, including for all the pre-simulation file checking and such.

settings_and_InputDataDir
The full path to where your input settings directory is.

SystemDataFilename
Name of the file for the system data. Just leaving this to the standard "SystemData.txt" and using the prepending approach has proved the most useful for me.

DIdataFilename
Name of the file for the astrometry data. Just leaving this to the standard "DIData.dat" and using the prepending approach has proved the most useful for me.

RVdataFilename
Name of the file for the radial velocity data. Just leaving this to the standard "RVData.dat" and using the prepending approach has proved the most useful for me.

outputData_dir
The full path to where the output data will be written. The simulator will create a sub-directory in here based on the "outputData_filenameRoot" setting.

outputData_filenameRoot
The root file name to use for the output folder and will also be used as part of the titles of the output plots and such.

RVonly
Only perform fitting to the radial velocity data? Set both this and "DIonly" to false to use 3D fitting.

DIonly
Only perform fitting to the astrometry data? Set both this and "RVonly" to false to use 3D fitting.

simAnneal
Run only Simulated Annealing? This is a good way to make sure that the Simulated Annealing stage is running well and finding a suitable starting place for the following MCMC simulations.

loopedMCMC
This might be still broken when you download SMODT, but it is designed to run bootstrap type simulations on a special version of the input data, maybe only the RV data. There are functions to

produce a set of the RV data with re-arranged errors. If many of these data sets are created, this version of the simulator would run on each data set and give different final solutions for each, that can all be plotted together to give an idea of the output solution set's range. It was suggested by some and thus implemented, but later given up on.

makePosteriorsPlot

Make plots of the posterior distributions? These will be histograms of the output values from Monte Carlo or MCMC, but not possible with Simulated Annealing by definition of how it runs.

makeOrbitPlots

Make plots of the orbits, both astrometry and radial velocity if requested with "RVonly" and "DIonly", with the data also represented. Tweaks to these plots can be made in the Python "plotToolbox.py" file if the user wishes.

makeSimAnnealProgPlots

Make progress plots of the Simulated Annealing run. This is a time series showing the values of all the varying parameters as a function of sample number. It is useful for seeing how the simulation converges to the final parameters.

startTemp

Starting temperature for the Simulated Annealing runs. Anything between 50-1000 could work, but it depends on how long you want to run that stage of the simulation for and how well constrained the parameters are.

delChainsAfter

Delete the output data files for each chain after the simulation is complete? This is a good way to save you from filling up your disk space when running many successive simulations. Remember that long simulations (over say 1e8) an cause large output data files on the order of hundreds of Gigabites.

delCombinedDataAfter

Delete the combined data file after simulation completes? After processing anything needed for the individual chains, they are combined to make a final single output data file (in the cases of MCMC and Monte Carlo only). Again, deleting these after can help save space.

TcStepping

The simulator calculates the Time of Last Periapsis (To) from other parameters and the Time of Center Transit (Tc), or vice versa. In the cases of very low eccentricities (under 0.3) it is recommended to step through the parameter space in Tc to help avoid biasing towards higher eccentric orbits. Thus, set this to true to do just that, or set it to false to calculate Tc from the varying To values.

CopyToDrobox

If you like, you can define the location of a DropBox directory, or another directory, you wish the smaller files produced by the simulator (plots, and maybe the log files) to be copied to at the end of the simulation. This is useful for showing collaborators recent results, or for you to see results on your home computer when running it on your work desktop.

CalcBurnIn

The burn in is the stage of the MCMC where the chain is "forgetting" its starting point. This is hopefully handled during the final sigma tunning at the end of the Simulated Annealing stage, but if it is a much larger value, then you should consider deleting the samples from the MCMC chains to remove any bias caused by the starting location.

calcCorrLengths

Another way to look at the progress of an MCMC chain is its correlation length. This will tell you how well the chain really explores the parameter space.

numSamples_SimAnneal

When running MCMC mode, this will set the number of samples for the Simulated Annealing stage.

makeMCMCprogPlots

Make progress plots of the MCMC chains. Just like the earlier setting for the Simulated Annealing chains, this can give you another visual way to investigate how the MCMC chains are exploring the parameter space. As MCMC is statistically designed to explore the space efficiently, this is commonly useless, but could pose useful to some users.

CalcGelmanRubin

The Gelman Rubin statistic (R) gives an indication on the simulations convergence to the final posterior distribution. R values of 1.0 indicate a perfect convergence, while values much higher indicate the simulation needs to be ran longer. Typically $R \leq 1.1$ is considered the criterion for convergence.

numTimesCalcGR

How often should the R values be calculate? This calculation is very heavy, so a smaller number is good, but remember you also want to know how the simulation is progressing as well. A happy middle would be ~ 100.

simulate_StarStar

Simulate a Star-Star system? ie. a binary.

simulate_StarPlanet

Simulate a Star-Planet system?

longAN_degMIN

Minimum value for the Longitude of the Ascending node parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

longAN_degMAX

Maximum value for the Longitude of the Ascending node parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

eMIN

Minimum value for the Eccentricity parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

eMAX

Maximum value for the Eccentricity parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

periodMIN

Minimum value for the Period parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

periodMAX

Maximum value for the Period parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

a_totalMIN

Minimum value for the total Semi-Major Axis parameter. The sum of both body's semi-major axes. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like. The masses of the stars, or planet, will be used to calculate their individual semi-major axes when performing radial velocity fitting. The a_total value can also be calculated from Kepler's third law, using the period and masses, so set MIN and MAX values to trigger this.

a_totalMAX

Maximum value for the total Semi-Major Axis parameter. The sum of both body's semi-major axes. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like. The masses of the stars, or planet, will be used to calculate their individual semi-major axes when performing radial velocity fitting. The a_total value can also be calculated from Kepler's third law, using the period and masses, so set MIN and MAX values to trigger this.

inclination_degMIN

Minimum value for the Inclination parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

inclination_degMAX

Maximum value for the Inclination parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

argPeri_degMIN

Minimum value for the Argument of Periapsis parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

argPeri_degMAX

Maximum value for the Argument of Periapsis parameter. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

T_Min

Minimum value for the Time of Last of Periapsis parameter, OR, the Time of Center Transit if Tc-Stepping=True. Zero for the MIN and MAX values indicate to not let it vary and instead take it as a fixed value from the SystemData.txt file. -1 indicates to use [earliestsEpoch-period,earliestEpoch], with earliestEpoch being the earliest observation in the data between the astrometry and RV data files. Or you could set it to very tight values around a fixed value if you like.

T_Max

Maximum value for the Time of Last of Periapsis parameter, OR, the Time of Center Transit if Tc-Stepping=True. Zero for the MIN and MAX values indicate to not let it vary and instead take it as a fixed value from the SystemData.txt file. -1 indicates to use [earliestsEpoch-period,earliestEpoch], with earliestEpoch being the earliest observation in the data between the astrometry and RV data files. Or you could set it to very tight values around a fixed value if you like.

K_MIN

Minimum value for the Radial Velocity Semi-Major Amplitude parameter. It will only vary if DIonly==false and inclination_degMAX==0. Else, it will be calculated from other values. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like. This value can be calculated from the masses, inclination and semi-major axes as well, so set the MIN and MAX values to zero to do calculate it that way instead.

K_MAX

Maximum value for the Radial Velocity Semi-Major Amplitude parameter. It will only vary if DIonly==false and inclination_degMAX==0. Else, it will be calculated from other values. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like. This value can be calculated from the masses, inclination and semi-major axes as well, so set the MIN and MAX values to zero to do calculate it that way instead.

RVoffsetMINs

Minimum value for the Radial Velocity Offset parameter. This will be a list, with one value for each RV data set provided. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

RVoffsetMAXs

Maximum value for the Radial Velocity Offset parameter. This will be a list, with one value for each RV data set provided. Zero for the MIN and MAX values indicate to not let it vary. Or you could set it to very tight values around a fixed value if you like.

3. Priors

In order to include previously determined information about the distribution some of the parameters tend to be for binary systems, sometimes non-uniform priors were used in the Metropolis-Hastings equation for the MCMC simulations.

As found by Duquennoy & Mayor (1991), for systems with periods≥1000 days, a function of f(e)=2e can be fit to their distribution, else no strong trend was seen. Thus, we adopt this result and use the normalized prior probability in Equation (7) for cases when the period is likely well over 1000days. Although, for those systems with close-in planets, the parametrizations $\sqrt{e}$ cos(ω) and $\sqrt{e}$ sin(ω) are used to allow for more efficient sampling and convergence when the eccentricity is very low (≤0.1), following the suggestion in Albrecht et al. (2012). This new parametrization result in a flat prior for e and ω.

$$P_n(e) = 2e \tag{7}$$

To take account for the random possible orbital orientations, a prior proportional to sin(i) is used, (8).

$$P_n(i) = \frac{\sin(i)}{\cos(i_{min}) - \cos(i_{max})} \tag{8}$$

For situations where the data of the orbit is sparsely sampled, such as in the case systems with periods over a month, one must be careful to avoid aliasing that can lead to a multitude of orbital period solutions. This effect is thoroughly discussed in Gregory (2005), and the suggestion of using a Jeffreys prior was described as the adequate solution to this problem, (9).

$$P_n(P) = \frac{1}{Pln(\frac{P_{max}}{P_{min}})} \tag{9}$$

Assuming the data is Gaussian distributed, the rejection function of the Metropolis-Hastings algorithm reduces to (10) once these normalized priors for orbits over 1000days are taken into account. For those cases where the orbit is well under 1000days, or if $\sqrt{e}\cos(\omega)$ and $\sqrt{e}\sin(\omega)$ are used, the ratio of the eccentricities in (10) simply becomes 1.

$$r(X_t, X_p) = max\left\{1, \frac{P_t}{P_p}\frac{e_p}{e_t}\frac{\sin(i_p)}{\sin(i_t)}e^{\frac{(\chi_t^2-\chi_p^2)}{2}}\right\} \tag{10}$$

where, χ^2 is the chi squared fit to the data given by (11).

$$\chi^2 \equiv \sum_{i=1}^{i=E} \frac{(model_i - observed_i)^2}{\sigma_i^2} \tag{11}$$

4. Models

4.1. True Anomaly Calculator

The True Anomaly Calculator is used for both models.

Table 1: Inputs to the True Anomaly Calculator.

Parameter	Description	Typical Range
t*	epoch of observation/image [julian date]	n/a
e	eccentricity of orbits [unitless]	[0.001,0.999]
T	Last Periapsis Epoch/time [julian date]	[t-period,t]
T_c*	Last Transit Center Epoch/time [julian date]	[t-period,t]
period	period of orbits [yrs]	[1.0,100.0]
verbose	Send prints to screen? [True/False](Default = False)	n/a

* = Normally measured/known (ie. not random numbers).

Table 2: Outputs of the True Anomaly Calculator.

Parameter	Description
n**	Mean Motion [rad/yr]
M**	Mean Anomaly [°]
E	Eccentric Anomaly [°]
θ	True Anomaly [°]

** = Not currently returned, but easily could be if needed.

First calculate the Mean Motion from the provided period:

$$n = \frac{2\pi}{period} \tag{12}$$

Use the Mean Motion (n), time of current epoch (t) and time of last periapsis (T) to get the Mean Anomaly (M):

$$M = n\left(\frac{(t-T)}{365.25} + phase\right) \tag{13}$$

where *phase* is the unit less phase offset, $(Tc - T)/365.25$, required when calculating the True Anomaly for Radial Velocity fitting to take into account that the velocity is zero at Tc, not T. The phase is set to zero for Astrometry fitting.

The relation between the Eccentric Anomaly (E) and the Mean Anomaly is given by Kepler's equation, and is a transcendental equation that must be solved using numerical methods.

$$M = E - e \times \sin(E) \tag{14}$$

In order to obtain the solution for E the fastest using the Newton's loop and (16), the closest guess of E should be used as the initial value of E′. This also helps to avoid ending up with one of the wrong solutions in cases where there are multiple crossings of the two functions that make up Equation (14). A suggested initial guess that we found to work well is given by (15), found by (Argyle 2004).

$$E_0 = M + e\sin(M) + \frac{e^2}{2M} sin(2M) \tag{15}$$

Newton's method to calculate E is then given by repeatedly calculating (16) in a loop:

$$E'' = E' - \frac{[E' - e \times \sin(E') - M]}{[1.0 - e \times \cos(E')]} \tag{16}$$

The loop completes when E'' and E' are the same to 10 decimal places. It is also checked to ensure the resulting value for E satisfies the original Equation (14) with the same precision. The maximum value of $e$possible was found to be 0.98, as rounding issues caused a division by zero above this.

Using the resultant E the True Anomaly (θ) can be calculated following:

$$\theta' = \cos^{-1}\left(\frac{[\cos(E) - e]}{[1.0 - e \times \cos(E)]}\right) \tag{17a}$$

$$\theta = \begin{cases} \theta' & \text{if E} \leq 180^\circ \\ 360^\circ - \theta' & \text{if E} > 180^\circ \end{cases} \tag{17b}$$

Equation (17a) has the unfortunate attribute that as the Eccentric Anomaly grows over 180°, the resulting value for the True Anomaly goes down, rather than up as should happen. This problem is rectified by applying the conditional statements of (17b).

The pictorial representation of these three anomalies is summarized in Figure 8.

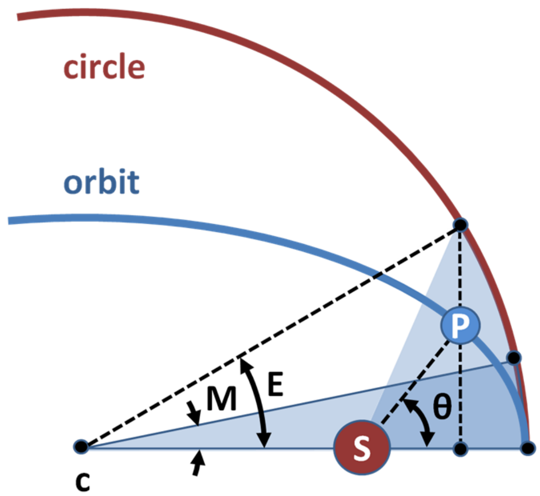

Fig. 8.— A diagram to compare the Eccentric, Mean and True anomalies of an orbiting planet. The planet is marked as P, the central is S, and the center of the ellipse is C in this plot.

4.2. Converting between T_o and T_{ij}

Used by the Radial Velocity model, not needed by the Astrometry model and is thus set to zero for those calculations.

In order to avoid biasing against circular solutions, one approach is to step through the parameter space using the Time of Inferior Conjunction (T_{ij}), instead of the Time of Periapsis (T_o). From the Earth's prospective, T_{ij} is when the companion object is located between the Earth and the primary (shown as T_{ij} in Figure 9), and T_o is when the companion crosses the plane of the sky moving away from the Earth, (Heintz 1978). In the radial velocity method, at T_{ij} the motion of the planet will be completely parallel to the plane of the sky, with no component along the line of sight to the Earth, resulting in a measured radial velocity of zero. By understanding these two definitions, it is possible to calculate one from the other if the values for the eccentricity, e, and argument of periapsis, ω, are known.

First, considering the plane of the sky is perpendicular to the line of sight (see Figures 5 &

9), it can be understood that the True Anomaly (θ_s) of the companion at the Time of Inferior Conjunction will equal 90°. Then:

$$\theta_s = 90° - \omega \tag{18}$$

When the companion passes in front of the primary directly along the line-of-site, this location is referred to as the Time of Center Transit (T_c). Although, in many cases the inclination of the system is not in the 90°+/-10° rough ranged needed, and no transit will be seen from Earth.

The Eccentric Anomaly of the companion can then be calculated from:

$$E_s = 2.0 * \tan^{-1}\left(\sqrt{\frac{1.0-e}{1.0+e}}\frac{\sin(\frac{\theta_s}{2.0})}{\cos(\frac{\theta_s}{2.0})}\right) \tag{19}$$

this equation is a manipulated version of that found in many texts, such as Heintz (1978), seen below.

$$\tan\left(\frac{\theta_s}{2.0}\right) = \sqrt{\frac{1.0+e}{1.0-e}}\tan\left(\frac{E_s}{2.0}\right) \tag{20}$$

The Kepler's Equation, where M_s is the Mean Anomaly of the secondary, (Heintz (1978) & Hilditch (2001)), is given by:

$$M_s = E_s - e \times \sin(E_s) \tag{21}$$

Knowing the relations for the Mean Motion n:

$$n = \frac{2\pi}{P} \tag{22a}$$

$$n = \frac{M_s}{t - T_o} = \frac{M_{ij} - M_o}{T_{ij} - T_o} \tag{22b}$$

where t is the epoch of observation and P is the orbital period. We can then apply these to get the following equation assuming M_o is zero.

$$\Delta t = t = \frac{M_{ij}P}{2\pi} \tag{23}$$

From this we can calculate either the Time of Inferior Conjunction, or Time of Periapsis knowing the other using:

$$T_{ij} = T_o + \Delta t \tag{24a}$$
$$T_o = T_{ij} - \Delta t \tag{24b}$$

Knowing the primary object is just 180° out of phase with the companion, these equations can be used for the primary replacing θ_p for θ_s in Equation (18):

$$\theta_p = 90^\circ - \omega + 180^\circ \tag{25}$$

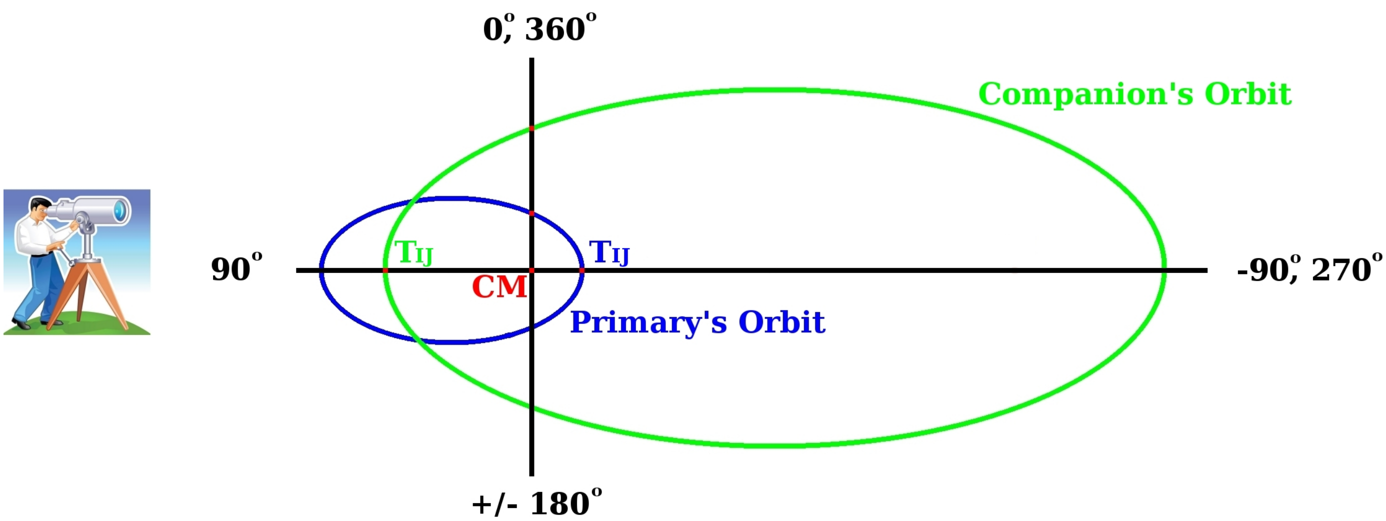

Fig. 9.— A diagram showing the location of the Inferior Conjunction locations (T_{ij} & T_{ij}) of two objects in a binary system, orbiting the center of mass (CM).

To take into account that T_{ij} can occur before or after T_o in a given system, the following conditionals are applied to get the correct value for Δt produced by Equation (23):

$$\Delta t = \begin{cases} \Delta t - P & \omega \text{ in } [\ 90^\circ{,}360^\circ],\ (\mathrm{T}_o > \mathrm{T}_{ij}) \\ \Delta t & \omega \text{ in } [-180^\circ{,}90^\circ],\ (\mathrm{T}_o < \mathrm{T}_{ij}) \end{cases} \tag{26}$$

these simplify to:

$$\Delta t = \begin{cases} \Delta t - P & \omega > 90^\circ,\ (\mathrm{T}_o > \mathrm{T}_{ij}) \\ \Delta t & \text{otherwise},\ (\mathrm{T}_o < \mathrm{T}_{ij}) \end{cases} \tag{27}$$

When considering the values for the primary object (triggered with the ‘backhalf’ boolean inside the function), these conditionals become:

$$\Delta t = \begin{cases} \Delta t - P & \omega \text{ in [-180°,270°], } (\mathrm{T}_o > \mathrm{T}_{ij}) \\ \Delta t & \omega \text{ in [270°,360°], } (\mathrm{T}_o < \mathrm{T}_{ij}) \end{cases} \tag{28}$$

simplifying to:

$$\Delta t = \begin{cases} \Delta t - P & \omega > 270°, \ (\mathrm{T}_o > \mathrm{T}_{ij}) \\ \Delta t & \text{otherwise, } (\mathrm{T}_o < \mathrm{T}_{ij}) \end{cases} \tag{29}$$

Therefore, with these equations, one only needs to step through the parameter space along one of them and calculate the other from it, along with e, P and ω.

4.3. Differences Between Investigating Stellar or Planetary Companion

NOTE: my recent investigations have found below to be essentially backwards!!! ie. put omega into DI and omega+180 into RV equations, using standard omega to calculate Tc !!!

The key difference when investigating a planetary compared to that of a stellar companion is the values of a and ω used in the astrometry model. As discussed in Schulze-Hartung et al. (2012) and Wright & Howard (2009), when the companion is a planet, the values of for the star are used, ie a_1 and ω_1. This is because the method was originally developed to find the orbit of the primary about the system's center of mass as the secondary was not seen. More recently though, in most cases the companion is seen and the astrometric values are for the location of the secondary relative to the primary instead of the center of mass. In these cases the apparent orbit is for the combined system and the values input into the astrometric model are a_1+a_2 and $\omega_2+\pi$, using the convention $\omega_1 = \omega_2 + \pi$.

One must be careful to take this into account when doing 3D simulations, to ensure that $\omega_2+\pi$ is passed into the astrometry model and simply ω_2 is used in the radial velocity one.

4.4. Direct Imaging (Astrometry) Model: Using the Thiele-Innes Method

The measurement of the astrometry values in the data, and the matching ones from the Thiele-Innes method applied here, are discussed in section 1.1.

The Thiele-Innes method to solve for the orbital elements of binary systems was first described in Thiele (1883), and later advanced with the inclusion of the Innes constants formulated in Van den Bos (1932). This approach has been mainstream ever since, and the equations to find the ephemeris are given below can be found in many text books, such as Aitken (1935), Binnendijk (1960) and Heintz (1978).

$$A = a[\cos(\Omega)\cos(\omega) - \sin(\Omega)\sin(\omega)\cos(i)] \tag{30a}$$

$$B = a[\sin(\Omega)\cos(\omega) + \cos(\Omega)\sin(\omega)\cos(i)] \tag{30b}$$

$$F = a[-\cos(\Omega)\sin(\omega) - \sin(\Omega)\cos(\omega)\cos(i)] \tag{30c}$$

$$G = a[-\sin(\Omega)\sin(\omega) + \cos(\Omega)\cos(\omega)\cos(i)] \tag{30d}$$

The x and y components of the location on the apparent ellipse are found with:

$$x_{TH-I} = AX + FY \tag{31a}$$

$$y_{TH-I} = BX + GY \tag{31b}$$

knowing,

$$X = \frac{r_j}{a}\cos(\nu_j) = \cos(E_j) - e \tag{32a}$$

$$Y = \frac{r_j}{a}\sin(\nu_j) = \sqrt{1-e^2}\sin(E_j) \tag{32b}$$

where, ν is the True Anomaly, E is the Eccentric Anomaly and r is the distance between the object and the primary at a particular time in the orbit; if the value of a is the semi-major axis of the secondary alone, instead of the combined system, then the value for r is the distance between the secondary and the center of mass. These are the only equations where the current position in the orbit is taken into account, and it is critical for making the predicted x, y values match the observed ones at a particular epoch (j).

As mentioned before, the respective x and y values can be calculated from the measured position angle (ϕ) and separation angle (ρ), in the units of [rads] and [$''$] respectively, given:

$$x_{TH-I} = \rho\cos(\phi) \tag{33a}$$

$$y_{TH-I} = \rho\sin(\phi) \tag{33b}$$

4.5. Radial Velocity Model

In the case of calculating the radial velocity residuals, there are various forms of the equation to calculate predicted radial velocity of the host star due to its companion's motion.

One very important thing that needs to be taken into account for radial velocity equations, that does not effect in direct imaging calculations, phase offset. This is the phase difference between the location of the companion at closest approach and inferior conjunction; in cases where the inclination is ∼90, the companion passes in front of the primary and the inferior conjunction is also referred to as the location of center transit. To take this into account, a manipulated version of the Mean Anomaly equation is used, which in turn effects the value of the True Anomaly used in the following radial velocity calculations.

Table 3: Inputs to the Radial Velocity Model.

Parameter	Description	Typical Range
t*	epoch of observation/image [julian date]	n/a
Sys_Dist*	measured system distance from Earth [PC]	[0.01,50.0]
i	inclination [°]	[0,180]
ω	Argument of Periapsis [°]	[0,360]
e	eccentricity of orbits [unitless]	[0.001,0.999]
T	Last Periapsis Epoch/time [julian date]	[t-period,t]
T_c*	Last Transit Center Epoch/time [julian date]	[t-period,t]
period	period of orbits [yrs]	[1.0,100.0]
a***	Total Semi-major axis [AU]	[0.1,200]
Mass1***	Mass of primary star [$M_\odot$]	[0.001,10]
Mass2***	Mass of companion [$M_\odot$]	[0.001, Mass1]
K***	Semi-major Amplitude of primary star [m/s]	[1,500]
verbose	Send prints to screen? [True/False](Default = False)	n/a

* = Normally measured/known (ie. not random numbers).

*** = Optional.

Table 4: Outputs of the Radial Velocity Model.

Parameter	Description
vr	Radial Velocity of primary due to companion [m/s]
K	Semi-major Amplitude of primary star [m/s]

Use the Mean Motion, n, time of current epoch (t), time of last periapsis (T), and time of transit center transit (Tc) to get the updated Mean Anomaly:

$$phase = \frac{Tc - T}{period_{days}} \tag{34a}$$

$$M = n\left[\frac{(t-T)}{365.25} + phase\right] \tag{34b}$$

$$\tag{34c}$$

The phase of the companion in Equation (34) is unit-less value between 0 and 1. In Equation (34b), the first term is essentially a fraction of how far the current epoch is away from the time of last periapsis multiplied by the Mean Motion throughout the orbit, resulting in units of radians. Following these equations (14)-(17b) would be used to calculate the True Anomaly needed in the radial velocity calculations shown below. This modified version of the Mean Anomaly equation is only needed when the eccentricity is non-zero, else Tc is set equal to T resulting in a zero phase, so this version reduces to the original form shown in (13).

The most common form of the radial velocity equation is:

$$vr = K[\cos(\theta + \omega_2) + e\cos(\omega_2)] \tag{35}$$

where K is the semi-major amplitude of the radial velocity curve.

The different formulations for calculating K are related to each other by Kepler's third law, given in Equation (6).

To find the radial velocity of the primary star due to companion's orbital motion:

$$K_s = \left[\frac{2\pi G}{P}\right]^{1/3} \frac{M_2}{M_1}(M_1 + M_2)^{1/3} \frac{\sin(i)}{\sqrt{1-e^2}} \tag{36a}$$

$$= \left[\frac{2\pi G(M_1 + M_2)}{P}\right]^{1/3} \frac{M_2}{M_1} \frac{\sin(i)}{\sqrt{1-e^2}} \tag{36b}$$

$$= \frac{2\pi a_1 \sin(i)}{P\sqrt{1-e^2}} \tag{36c}$$

When the companion is a planet, it can commonly be assumed that $\text{Mass}_{planet} \ll \text{Mass}_{star}$, so $M_1 + M_2 \approx M_1$. This simplifies the equation for K to:

$$K_s = \left[\frac{2\pi G}{P}\right]^{1/3} \frac{M_2 \sin(i)}{M_1^{2/3}} \frac{1}{\sqrt{1-e^2}} \tag{37}$$

As the mass of each object is needed to calculate the semi-major axis of the primary, a_1, using (36c) keeps things simple making this the favored form.

Overall, this leads to 3 different ways to calculate the radial velocity of the primary due to a companion:

$$vr_c \text{ or } vr_p = \frac{2\pi a_1 \sin(i)}{P\sqrt{1-e^2}}[\cos(\theta+\omega_2)+e\cos(\omega_2)] \tag{38a}$$

$$vr_c = \left[\frac{2\pi G(M_1+M_2)}{P}\right]^{1/3}\frac{M_2}{M_1}\frac{\sin(i)}{\sqrt{1-e^2}}[\cos(\theta+\omega_2)+e\cos(\omega_2)] \tag{38b}$$

$$vr_p = \left[\frac{2\pi G}{P}\right]^{1/3}\frac{M_2\sin(i)}{M_1^{2/3}}\frac{1}{\sqrt{1-e^2}}[\cos(\theta+\omega_2)+e\cos(\omega_2)] \tag{38c}$$

where vr_c would be the radial velocity of the primary star due to a companion star, and vr_p due to a companion planet around the same primary. In the SMODT code, that of (38a) is referred to as "vrCalculatorSemiMajorType", (38b) as "vrCalculatorMassType", and (38c) as "vrCalculatorPlanetMassType", with the default in all cases being (38a).

Finally the χ^2 is calculated following:

$$\chi^2 \equiv \sum_{i=1}^{i=E}\frac{(model_i - observed_i)^2}{\sigma_i^2} = \sum_{i=1}^{i=E}\frac{[(vr_c+vr_p)-(RV_{data}-\gamma)]^2}{\sigma_i^2} \tag{39}$$

where γ is the velocity offset of that instrument. In the case that the velocity of the system's center of mass WRT the Earth has not been removed, the observed component in (39) becomes $(RV_{data} - \gamma_{Instrument} - \gamma_{COM})$ (Paddock (1913) & Schulze-Hartung et al. (2012)). When only investigating the radial velocity fit of a single companion, the other would naturally be zero in the above equation.

REFERENCES

Aitken, R. G. 1935, The Binary Stars, McGraw-Hill Book company (McGraw-Hill)

Albrecht, S., Winn, J. N., Butler, R. P., Crane, J. D., Shectman, S. A., Thompson, I. B., Hirano, T., & Wittenmyer, R. A. 2012, ApJ, 744, 189

Argyle, B. 2004, Observing and Measuring Visual Double Stars, Patrick Moore's Practical Astronomy Series No. v. 1 (Springer)

Batten, A. 1973, Binary and multiple systems of stars, International series of monographs in natural philosophy (Pergamon Press)

Binnendijk, L. 1960, Properties of Double Stars (University of Pennsylvania Press)

Duquennoy, A., & Mayor, M. 1991, A&A, 248, 485

Gregory, P. C. 2005, ApJ, 631, 1198

Heintz, W. 1978, Double stars, Geophysics and astrophysics monographs (D. Reidel Pub., Co.)

Hilditch, R. 2001, An Introduction to Close Binary Stars, Cambridge astrophysics series (Cambridge University Press)

Paddock, G. F. 1913, PASP, 25, 208

Schulze-Hartung, T., Launhardt, R., & Henning, T. 2012, ArXiv e-prints

Thiele, T. 1883, Astronomische Nachrichten, 104

Van den Bos, W. H. 1932, Circular of the Union Observatory Johannesburg, 86, 261

Wright, J. T., & Howard, A. W. 2009, ApJS, 182, 205

This preprint was prepared with the AAS LaTeX macros v5.2.