Investigating stability of triple and quadruple systems

Diganta Bandopadhyay, Supervisor: Silvia Toonen

July 2019

Abstract

In this study we present a simulation code[1] that can be used to investigate the stability of triple and quadruple systems. We compare the stability of systems produced by the OBin (Kouwenhoven et al. [2007]) and T14 (Tokovinin [2014]) orbital separation distributions across triple and quadruple stellar systems. The OBin model is found to produce more interacting binary systems in both the circular orbit and thermal eccentricity cases (for triple stellar systems). We find the thermal eccentricity model produces systems with lower distances of closest approach between the triple and inner binary stars (compared to the circular case), resulting in more dynamical interactions between the triple and inner binary stars. In the circular orbit case we find the T14 model produces more systems than the OBin model where the 3 body dynamical interactions between the inner binary and triple star are significant. We compare the stability of the mass ratio distributions; $\frac{1}{q}$ and uniform, both mass ratio distributions result in similar numbers of stable systems due to the weak dependence of the stability criteria on mass ratio. We find no significant mass ratio bias within triple or quadruple systems for the uniform mass ratio model. The $\frac{1}{Q}$ mass ratio model produces a stable distribution biased towards low mass ratios. The quadruple OBin system is found to produce more interacting inner binary systems than the triple case. Similarly to the triple case, in the T14 model the 4 body dynamical interactions between the tertiary, quaternary and inner binary stars become significant. Overall we find lower eccentricity systems to be more stable in the quadruple and and triple cases, however in the quadruple case we find similar number of systems with any given eccentricity.

1 Introduction

A significant proportion of stars are located in stellar systems with multiple companions. The multiplicity fraction is the probability of a stellar system to contain multiple stars. The multiplicity fraction for solar type main sequence primaries is 0.5 (Moe and Di Stefano [2016]). Figure 1 shows the variation of stellar companion fraction (up to quadruple systems) as a function of primary mass.

In this study we set $M1 = 2M_{\odot}$ (code presented can be easily changed to investigate other primary masses). At this primary mass, we find a triple and binary fraction respectively of approximately 0.10 and 0.35. There is an insignificant number of quadruples found in the $M1 = 2M_{\odot}$ region, however the number of quadruples grows with primary mass. The general observed trend in figure 1 is multiplicity fraction increases with primary mass. Higher mass primary stars are able to more strongly (gravitationally) bind outer stars (at larger orbital radii), therefore the multiplicity fraction increases.

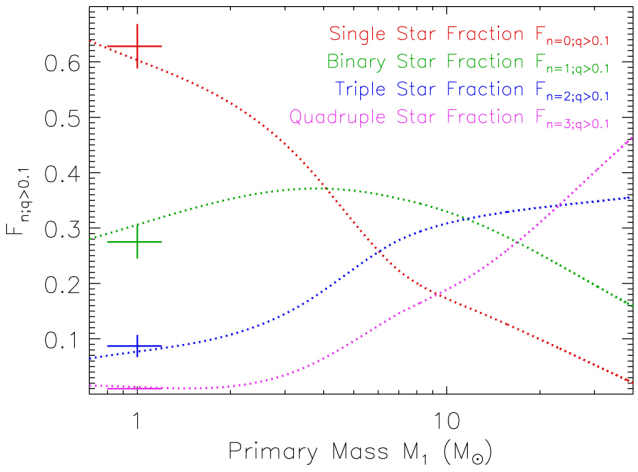

Figure 1: Single, binary, triple and quadruple star fractions as a function of primary mass in stellar systems (Moe and Di Stefano [2016])

In this study we investigate the stability of triple and quadruple systems across orbital elements. Systems found to be unstable are weakly bound, therefore likely to undergo external interactions resulting in dis-

[1] https://github.com/dig07/Summer-project-triple-and-quadruple-stability

association within dynamical time, so we are unlikely to observe these systems. We investigate the possibility of any biases within the stable distributions of each orbital element. For the triple systems we are interested in the following:

- Differences between the OBin (Kouwenhoven et al. [2007]) and T14 (Tokovinin [2014]) orbital separation distributions.
- Difference in stability between the uniform (Moe and Di Stefano [2016]) and $\frac{1}{q}$ (Trimble [1990]) mass ratio distributions.
- Difference in stability between the circular orbit assumption and the thermal eccentricity distribution.

For the quadruple system we also test the differences between the OBin and T14 Orbital separation distribution. Additionally we are interested in differences in the stability distributions (of different orbital elements) compared to the stability distributions produced by the triple systems.

2 Triples

We create synthetic populations of triple stellar systems (from here on out called triples) using defined probability density functions for the orbit parameters; orbital separation (inner and outer orbits), mass ratio, inclination and eccentricities. Each unique system configuration is tested for stability against the criterion presented in Mardling and Aarseth [2001].

$$\left.\frac{a_{out}}{a_{in}}\right|_{crit} = \frac{2.8}{1-e_{out}}\left(1-\frac{0.3i}{\pi}\right)\left(\frac{(1+q_{out})(1+e_{out})}{\sqrt{1-e_{out}}}\right)^{\frac{2}{5}} \quad (1)$$

The stability of the population is plotted to show the parent and stable (system) distributions (histograms) across variables of interest. We assume a hierarchical triple system of the form shown in figure 2. S1 and S2 refer to the stars within the inner binary, S3 refers to outer star orbiting the inner binary COM[2].

Figure 2: Triple Hierarchical System

We define the following parameters for our simulations:

- a_{in} : Orbital separation of the two stars within the inner binary (S1 & S2).
- a_{out} : Orbital separation between the COM of the inner binary and the outer orbiting star (S3).
- e_{out} : Eccentricity of outer orbit.
- Inclination : Inclination of orbital plane of tertiary star with respect to orbital plane of inner binary.
- q_{out} (Outer Mass ratio) : Outer Mass ratio is defined as $\frac{M3}{M1+M2}$.

[2]Centre of mass

2.1 Orbit parameter populations

Populations of triple star systems are constructed by drawing orbital parameters from appropriate distributions. Some parameters have multiple (currently) hypothesised distributions they could originate from, in these cases we run simulations with both distributions to reveal differences between them in the context of stability. The distributions we consider are summarised in the Stellar Multiplicity review by Duchêne and Kraus [2013].

2.1.1 Orbital Separation

In this study, we consider the OBin and T14 orbital separation distributions. The OBin model is based on our understanding of observed primordial binary systems. The T14 distribution is motivated by triple samples suggested by Tokovinin [2014]. For the OBin model, we adopt a uniform distribution in log of orbital separation (Kouwenhoven et al. [2007]), in the range $[5, 5\times10^6]R_\odot$. A histogram of orbital separations picked from OBin distribution is shown in figure 3.

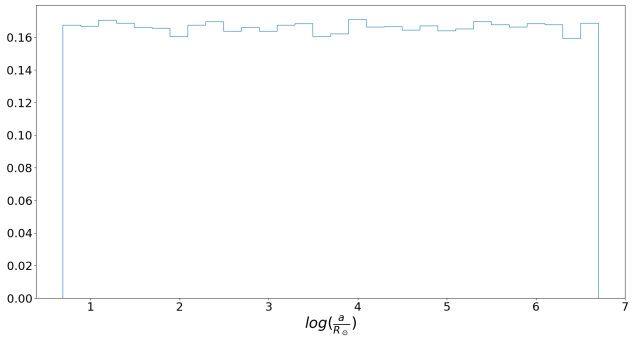

Figure 3: Histogram of orbital separations drawn from OBin model PDF.

For the T14 model we adopt a log normal distribution in orbital period with parameters $\mu = 5, \sigma = 2.3$ (days) as presented in Tokovinin [2014]. Note unlike the OBin model, we don't impose a high/low cutoff for

the orbital separation. A histogram of orbital separations picked from T14 distribution is shown in figure 4.

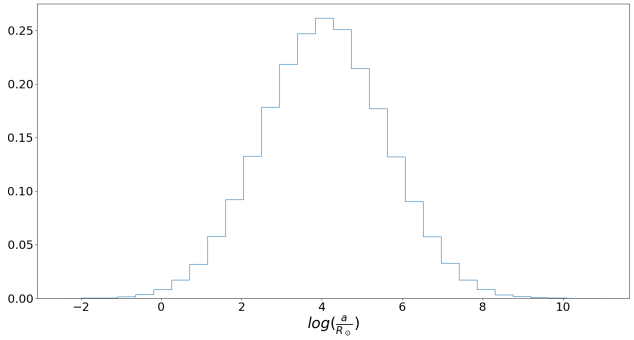

Figure 4: Histogram of orbital separations drawn from T14 model PDF.

Sorting the orbital separations (using algorithm 2 in section 2.1.2) into the a_{in} and a_{out} arrays produces the histograms shown in figures 5 and 6 respectively for the OBin and T14 distributions.

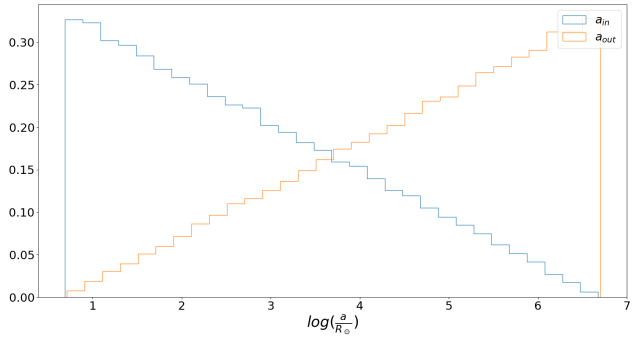

Figure 5: Histogram of a_{in} & a_{out} arrays drawn and sorted from the OBin distribution.

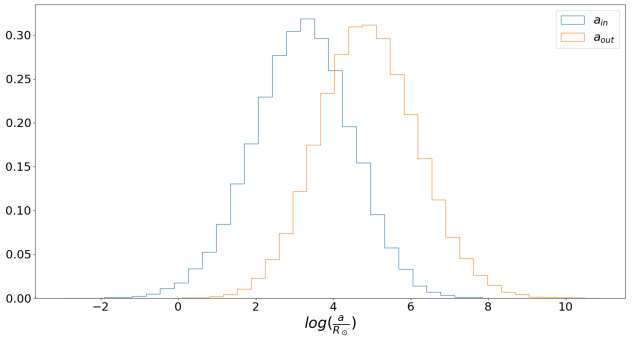

Figure 6: Histogram of a_{in} & a_{out} arrays drawn and sorted from the T14 distribution.

2.1.2 Sorting

We consider two algorithms to sort the drawn orbital separations into a_{in} & a_{out} arrays. Both the algorithms use (as an input) an array of randomly drawn orbital separations from the initial distributions shown in figures 3 and 4.

Sorting algorithm 1

- Choose the first element of the input array, set this as a_{in}.
- Filter the array to find the sub-set of elements larger than a_{in}. Randomly choose an element from this set, assigning this element as a_{out}.
- Append a_{in} and a_{out} to their respective lists.
- Delete a_{in} and a_{out} from the input array (to avoid repeats of the same elements).
- Repeat previous steps until array is empty

Sorting algorithm 2

- Select first two elements of the input array of orbital separations, set larger and smaller elements to a_{out} and a_{in} respectively and append to their respective lists.
- Delete first two elements of input array. (Since these are the a_{in} and a_{out} we just assigned.
- Repeat previous steps until array is empty.

Using sorting algorithm 1 produces a slightly biased parent distribution, whereas algorithm 2 produces a completely unbiased distribution. Figures 7 and 8 respectively plot the parent distributions for the OBin and T14 models, created by algorithms 1 and 2. Since a_{in} and a_{out} are selected and sorted "randomly" from the same PDF (which is symmetric) we expect a symmetric histogram. Algorithm 1 is slightly biased due to the cutoff we impose within the initial array that all possible choices for a_{out} have to be larger than a_{in}. Note interestingly using algorithm 1 biases the OBin and T14 respectively towards high a_{out} and low a_{in}.

For this study we use algorithm 2 to sort our orbital separations for triples into a_{in} & a_{out} arrays. ‘

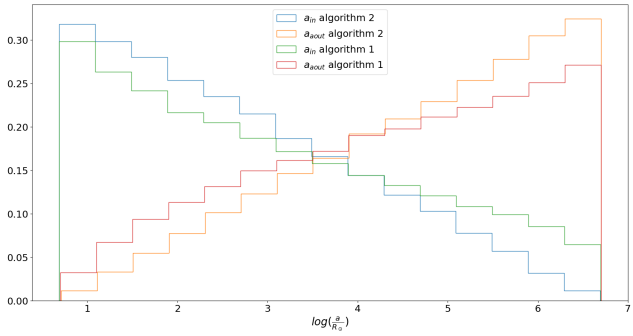

Figure 7: Histogram of a_{in} & a_{out} arrays drawn and sorted from the OBin distribution.

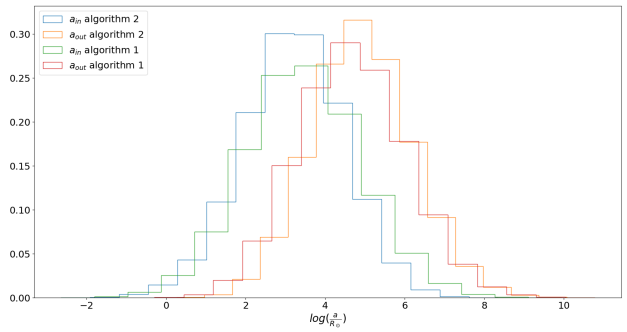

Figure 8: Histogram of a_{in} & a_{out} arrays drawn and sorted from the T14 distribution.

Figures 9 and 10 show the parental distributions created by reversing the order of algorithm 1 to; pick a_{out} and select a_{in} from subset of input array with elements lower than a_{out}.

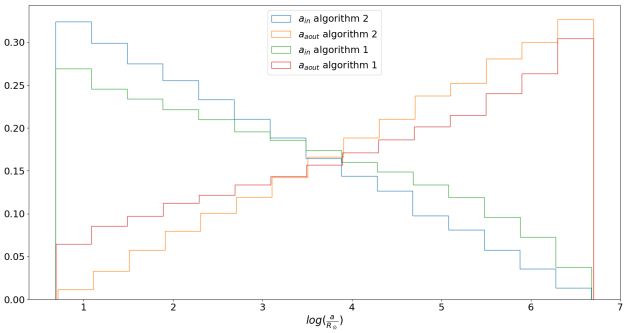

Figure 9: Histogram of a_{in} & a_{out} arrays drawn and sorted from the OBin distribution (using algorithm 2 and reversed version of algorithm 1).

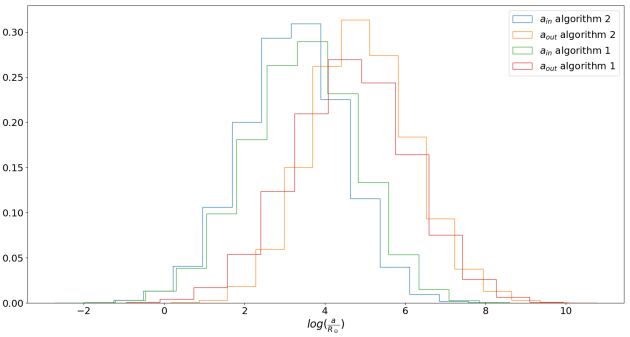

Figure 10: Histogram of a_{in} & a_{out} arrays drawn and sorted from the T14 distribution (using algorithm 2 and reversed version of algorithm 1).

We use algorithm 2 in our study. However within algorithm 2 we delete elements from the input array, this could potentially introduce biases which are not easily visible directly from the results in figures 7 and 8. We introduce sorting algorithm 3 which is free of biases as it directly draws values from the distribution for each iteration (therefore no biases are introduced due to deleting elements from the input array). Figures 11 and 12 compare the distributions produced by algorithm 2 (used for study) and algorithm 3. We observe similar distributions produced by both algorithms, therefore we can conclude there are no significant biases within algorithm 2. However we recommend algorithm 3 for future studies as it is simpler to implement and guarantees the lack of biases within the distributions.

Sorting algorithm 3

- Instead of using the input array we directly draw 2 values from the probability distribution.
- Set smaller value to a_{in} and larger value to a_{out}, append to respective lists.
- Repeat previous steps until desired number of orbital configurations is reached.

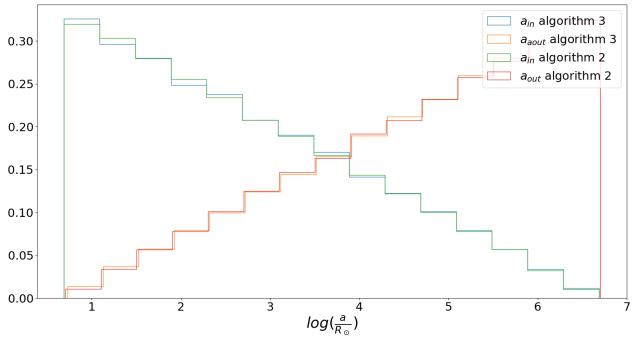

Figure 11: Histogram of a_{in} & a_{out} arrays drawn and sorted from the OBin distribution using algorithms 2 and 3

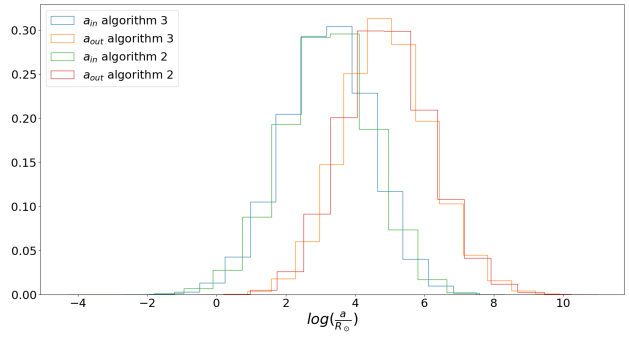

Figure 12: Histogram of a_{in} & a_{out} arrays drawn and sorted from the T14 distribution using algorithms 2 and 3

2.1.3 Mass Ratio

We trial two mass ratio distributions; flat (Moe and Di Stefano [2016]) and $\frac{1}{q}$ (Trimble [1990]), histograms

of mass ratios drawn from these distributions are shown in figure 13. We define $M1 = 2M_{\odot}$. In the triple section we directly draw the q_{out} distributions, without explicitly calculating $M1, M2$ or $M3$. The algorithm to draw quadruple mass ratios is explained in section 3.1.

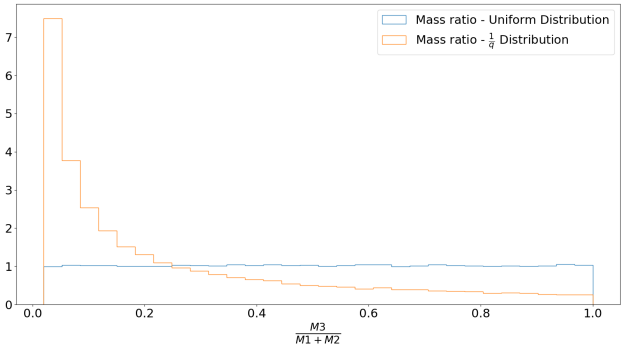

Figure 13: Histograms of mass ratios drawn from $\frac{1}{q}$ and flat distributions

2.1.4 Eccentricity

We assume a thermal distribution in eccentricity (Heggie [1975]). The thermal distribution we define is appropriate for a dynamically relaxed thermal population and is expected to be $f(e) = 2e$. We assume distributions of mass ratio and eccentricity are independent. Figures 14 and 15 respectively show the CDF[3] and a histogram of eccentricities randomly drawn from the thermal distribution.

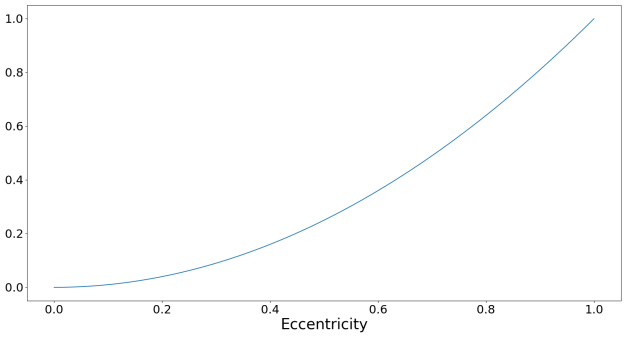

Figure 14: Cumulative distribution for thermal eccentricity model $f(e) = 2e$.

[3]Cumulative distribution function

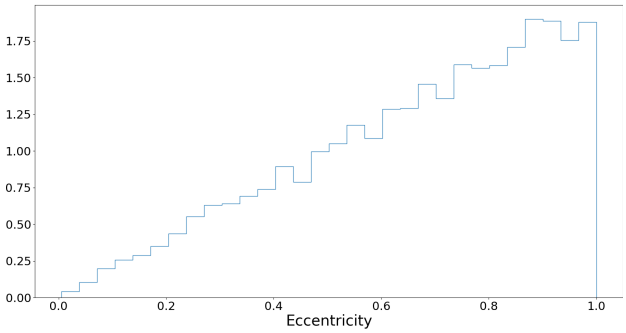

Figure 15: Histogram of eccentricities drawn from thermal distribution.

2.1.5 Inclination

We assume a flat distribution in the cosine of the inclination angle, histogram of inclinations drawn from this distribution is shown in figure 16.

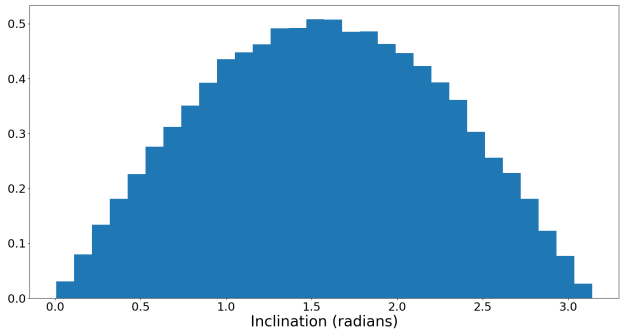

Figure 16: Histogram of drawn inclinations.

2.2 Code

The code for triple simulation is written in Python and FORTRAN 95. FORTRAN 95 is used for the computationally intensive section, due to its higher efficiency (compared to python).The "Initial_PDF.py" script is used to generate the parent distributions for all the orbit parameters, these are saved onto a text file.

The parent distributions text file is read into the Python simulation script and arrays for each of the parameters are initialised. In order to test stability, we pass every possible configuration of orbits into equation 1, and track the stability of each parameter across its respective range. To cover the entire parameter space the simplest algorithm is a quadruple nested for loop, the layers of this loop iterate over the following arrays:

- **a_ratio**: Passing in the array of a_{out} and a_{in}, the array a_ratio can be calculated as $a\_ratio = \frac{a_{out}}{a_{in}}$.

- **mass ratios**: Array of mass ratios we test over.
- **eccentricities** : Array of eccentricities we test over.
- **inclinations** : Array of inclinations we test over.

With the innermost layer of the for loop containing the stability test for that unique orbit configuration. However this algorithm is extremely inefficient and cannot realistically (within reasonable execution time) use large parameter arrays, resulting in the final distributions showing no clear trends (due to low number statistics).

To reduce the computation time, we use multiprocessing, vectorised operations and FORTRAN 95. We split up the parameter space by *a_ratio*, and distribute the computation between the cores (using the multiprocessing library). We use the pool.Map() function from the multiprocessing library to assign parts of the parameter space to each process (on each core). Each process is initialised with a single element of *a_ratio* array and the mass ratio, eccentricity, and inclination arrays.

We choose to use FORTRAN 95 to run the nested for loops due to its high (comparative) efficiency. The FORTRAN code is compiled into a Python library (using f2py), which is imported into the simulation script and called by each individual process.

2.2.1 FORTRAN Algorithm

Algorithm 1 shows the pseudo-code for the FORTRAN library. Each process initialised from the python simulation script is assigned one unique *a_ratio* (which is pre-calculated). The processes each initialise the FORTRAN library with a unique *a_ratio* and the full arrays of mass ratios, eccentricities and inclinations. Within the "Simulation" procedure e, i, q respectively correspond to the eccentricity, inclination and mass ratio arrays. Counter arrays are set up for the eccentricities and mass ratios, the arrays contain the counts of stable orbits at each configuration. The counter variable for the *a_ratio* is an integer due to each process (and so each instance of the FORTRAN library) computing using the parameter space covered by one *a_ratio*. The "Stable" procedure uses vectorised array operations to efficiently count number of stable orbits with a unique *a_ratio*, mass ratio and eccentricity. The parameters passed into this procedure are all floats, except the inclination array. The "inclination_limit" variable uses a rearranged form of equation 1 to calculate the minimum inclination for which this set of systems are stable. The "where" function uses vectorised array operations to count the instances where the inclination is larger than "inclination_limit" minimum limit. The number of stable systems is then assigned to the variable "Num" which is used to increment counter variables for the parameters. Finally the FORTRAN library returns the integer counter variable (*a_ratio*) and counter arrays to the Python simulation script.

2.2.2 Python data handling

The Python simulation script contains 3 arrays initialised using the "multiprocessing" library, these arrays are shared data-structures between the processes. These arrays are used to collect and combine the results from each process. The array for measuring stability of *a_ratio* contains the the integer counter variable (returned from the FORTRAN library) inserted in the same index as its corresponding *a_ratio*. Since the full mass ratio and eccentricity arrays are passed into each process, the FORTRAN script returns arrays containing counter variables for each eccentricity and mass ratio, these are each added (element-wise) to their corresponding "multiprocessing" counter arrays.

One of the plots we want to create is not directly linked to an orbital parameter. The plot over $a(1-e)$ shows the closest distance of approach (between the two objects being considered). We can create this plot for $a_{out}(1-e_{out})$ since we are able to measure the stability for e_{out}, to generate the data for this plot, we collect the eccentricity counter arrays (in a list) from each process in the order they were executed. The processes are initialised in the order within the *a_ratio* array, and the *a_ratio* array has the same ordering as the a_{out} array. Therefore we can calculate $a(1-e)$ by looping over the eccentricity array for each a_{out} element, collecting the eccentricity counter arrays (in the order of *a_ratio*) therefore serves as a counter for the closest approach variable. We create the plot of $a_{in}(1-e_{in})$ by using the counters from the orbital separation (a_{in}) stability arrays. Since e_{in} is not part of the stability criteria given in equation 1, we are not able to directly measure the stability of $a_{in}(1-e_{in})$. To create the inner binary closest approach plot, we assume inner binary has a thermal distribution of eccentricity, we use the stability of each a_{in} to work out the stability of each $a_{in}(1-e_{in})$ (Since all e_{in} are stable as stability criteria is independent of inner eccentricity). We also create the plot of $a_{in}(1+e_{in})$ similarly.

2.3 General results

The following subsections contain results showing the parent and stable systems across the variables: orbital separation, eccentricity, mass ratios and perapsis distance. We consider orbital configurations drawn from both the OBin and T14 model. We assume a uniform distribution in mass ratios in the following

Algorithm 1 FORTRAN library

procedure STABLE(a_ratio,eccentricity,inclinations,q_ratio)
    Types: a_ratio: float, eccentricity: float, inclinations: array, q_ratio: float
    inclination_limit = $\frac{\pi}{0.3}\left(1-\frac{a\_ratio}{\left(\frac{2.8}{1+eccentricity}\right)\left(\frac{(1+q\_ratio)(1+eccentricity)}{\sqrt{1-eccentricity}}\right)^{\frac{2}{5}}}\right)$
    `stable_sys = length(where(inclinations>inclination_limit))`
    `Return stable_sys`
procedure SIMULATION(a_ratio,eccentricities,inclinations,q_ratios)
    Types: a_ratio: float, eccentricities: array, inclinations: array, q_ratios: array
    eccentricities_counter_array = array([0]×length(eccentricities))
    q_ratio_counter_array= array([0]×length(q_ratios))
    a_ratio_counter = 0
    for q_index in range(length(q_ratios)) **do**
        q_ratio = q_ratios[q_index]
        `q_counter=0`
        for eccentricity_index in range(length(eccentricities)) **do**
            eccentricity = eccentricities[eccentricity_index]
            `Num=Stable(a_ratio,eccentricity,inclinations,q_ratio)`
            `a_ratio_counter+=Num`
            `q_counter+=Num`
            `eccentricities_counter_array[eccentricity_index] +=Num`
        `q_ratio_counter_array[q_index] += q_counter`
    `Return a_ratio_counter,q_ratio_counter,eccentricities_counter_array`

plots (choice of mass ratio distribution does not make a significant difference, see section 2.5).

2.3.1 OBin

In the following list we detail the sample size of each relevant orbital element. A sample size of 100 corresponds to an array with 100 different values drawn from the appropriate distribution. In the following simulation we have 10000 different orbital separation pairs, 1000 different mass ratios and 100 different eccentricities and inclinations.

Simulation details:

- Sample size for orbital separations: 10000 unique orbital a_{in} a_{out} pairs.
- Sample size for mass ratios: 1000.
- Sample size for eccentricities: 100.
- Samples size for inclinations: 100
- Total number of unique orbits: 10^{11}

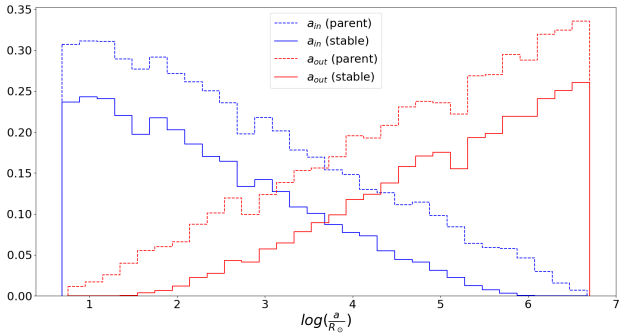

Figure 17: Histogram showing parent and stable orbital separation distributions for OBin model

For the OBin orbital separation model (Figure 17), systems with lower a_{in} and higher a_{out} are found to be more stable. Maximum stability for a_{in} and a_{out} respectively are found at the minimum and maximum of the orbital seperation range. The maximum stable a_{in} has a noticeable decrease between the parent and stable distribution, while the minimum stable a_{out} has a noticeable increase between the parent and stable distributions.

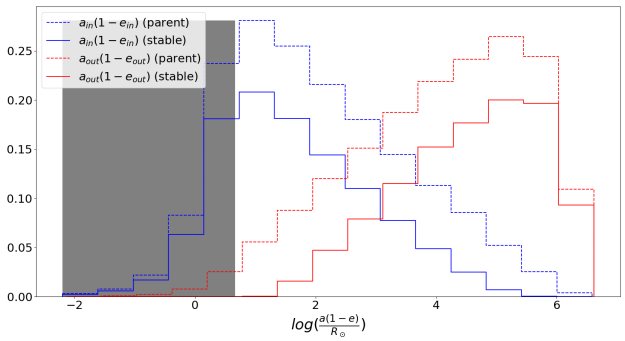

Figure 18: Histogram showing the parent and stable periapsis distance distributions for OBin model

The shaded region in figure 18 is defined as $a(1-e) < 5R_\odot$, due to the very low periapsis of systems in this region, the two bodies being considered would overlap, therefore these systems are not stable in the context of long term evolution. The systems just outside this region could potentially undergo mass transfer (the exact nature of the mass transfer depends upon the mass ratio of the bodies being considered). The inner binary periapsis distance peaks close to (but not within) the shaded region, implying the OBin model produces significant numbers of stable triple systems with interacting binaries. No systems are found with a_{out} in this region, however significant numbers of systems are found with a_{in} below this limit. A similar trend to figure 17 is observed, the maximum stable periapsis distance for $a_{in}(1-e_{in})$ has a noticeable decrease from the parent to stable distribution, whereas the minimum stable $a_{out}(1-e_{out})$ has a noticeable increase from parent to stable distribution. The offset[4] between the stable and parent curves is roughly constant for both a_{in} and a_{out} curves in the moderate periapsis distance region. However both the a_{in} and a_{out} respectively have small offsets in the extremely low and high periapsis distances. Systems are pushed to have lower inner binary and higher tertiary star periapsis distances.

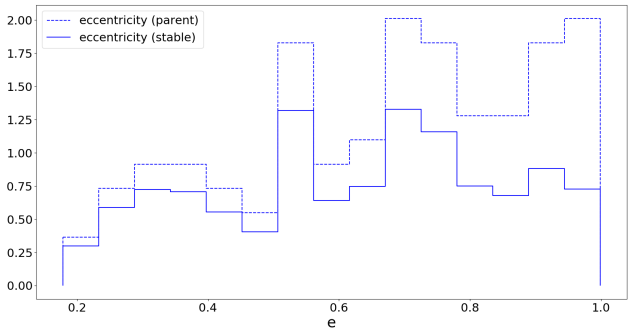

Figure 19: Histogram showing the parent and stable eccentricity distributions for the OBin model

The offset between the parent and stable eccentricity grows with eccentricity (Figure 19). Overall low eccentricity configurations are seen to more stable (comparing parent and stable curves proportionally for each bin).

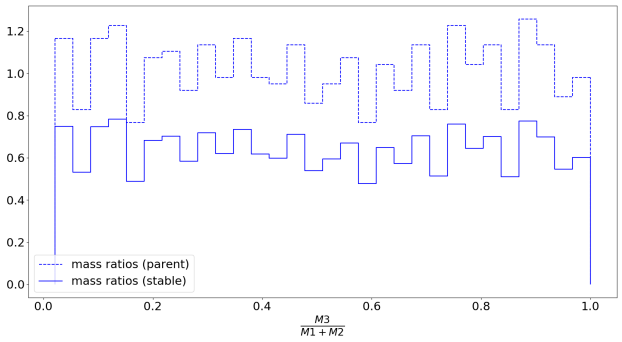

Figure 20: Histogram showing the parent and stable mass ratio distributions for the OBin model

The offset between the parent and stable distributions is constant throughout the entire range of figure 20. There is no outer mass ratio bias in the stable systems found for the OBin model.

2.3.2 T14

Simulation details:

- Sample size for orbital separations: 10000 unique orbital a_{in} a_{out} pairs.
- Sample size for mass ratios: 1000.
- Sample size for eccentricities: 100.
- Samples size for inclinations: 100
- Total number of unique orbits: 10^{11}

[4]When we discuss the offset in this study, unless explicitly stated otherwise we consider the offset for each bin (in probability) between the parent and stable distributions. The variability of the offset describes how the offset between the parent and stable distributions changes throughout the orbital element range

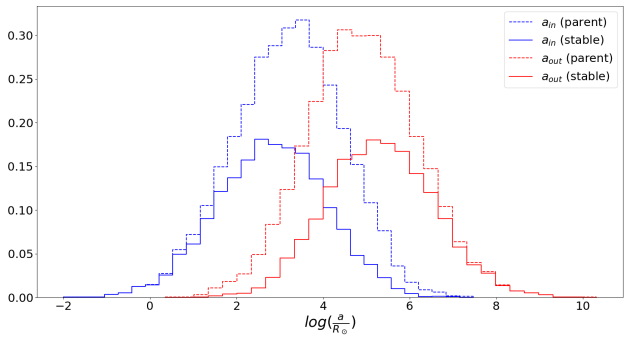

Figure 21: Histogram showing parent and stable orbital separation distributions for T14 model

The T14 orbital separation model (Figure 21) populates mainly the moderate orbital separation region, while the OBin model (Figure 17) mainly populates the extreme high and low orbital separations (parent distribution). The offset between the parent and stable curves varies across the range of orbital separations significantly more in the T14 model than in the OBin model. While the offset in the OBin distribution is seen to remain constant through the entire plot, we find in the T14 model, the offset between the distributions is maximum at the peaks of the stable distribution curves. The peaks of the stable distributions in the T14 model are also pushed apart implying the inner binary is pushed towards lower orbital separation, whereas the outer binary is pushed towards higher orbital separation (similar to result found for OBin case). The offset in maximum a_{in} and minimum a_{out} (between stable and parent distributions) is found to be lower for the T14 distribution than for the OBin distribution.

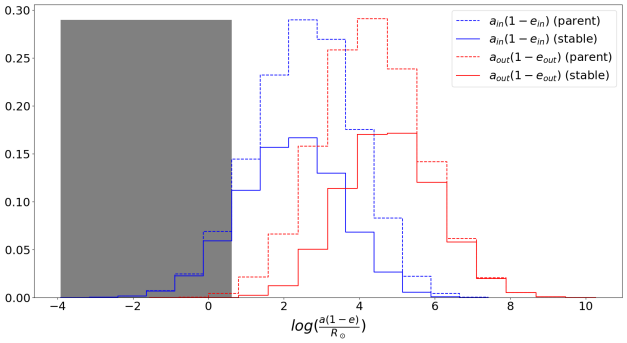

Figure 22: Histogram showing the parent and stable periapsis distance distributions for T14 model

The offset in figure 22 follows the same trend as the orbital separation distribution (Figure 21), largest offset corresponds to the peak of the stable curves and there is a larger offset in the moderate periapsis distance region (inner region) of both curves. Compared to the periapsis distance plot for the OBin model (Figure 18 we find a lower offset (between stable and parent distributions) in the maximum inner orbit periapsis distance and minimum outer orbit periapsis distance. The T14 model produces less systems in the region $a(1-e) < 5R_\odot$ (for the inner binary) than the OBin model. The periapsis distance for the T14 model peaks significantly further from the shaded region than in the OBin case. We can conclude the OBin model produces more interacting binary systems (stable) than the T14 model.

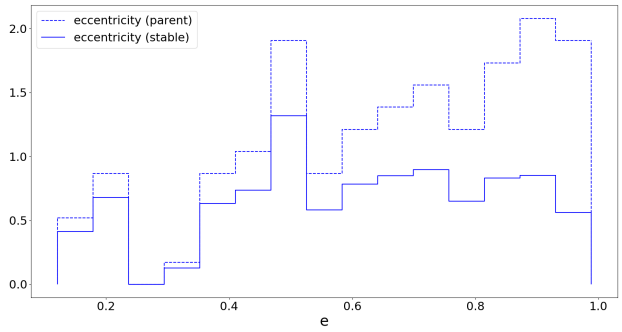

Figure 23: Histogram showing the parent and stable eccentricity distributions for the T14 model

The eccentricity distribution for the T14 model (Figure 23) shows a similar trend as the OBin model distribution (Figure 19). Overall the same trend of lower eccentricities are more stable is found.

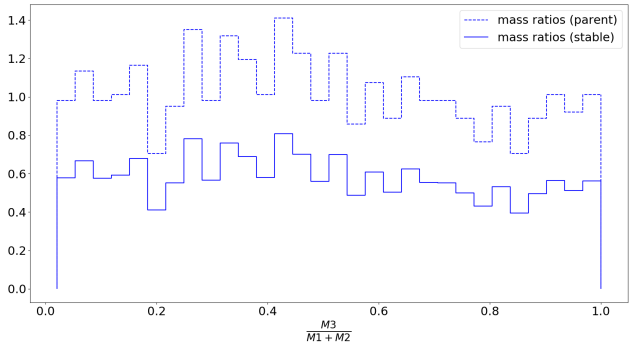

Figure 24: Histogram showing the parent and stable mass ratio distributions for the T14 model

The mass ratio distribution for the T14 model (Figure 24) has uniform offsets between the parent and stable distribution through the entire plot. This is the same result found for the OBin mas distribution (Figure 20). No bias found for outer mass ratio distribution of systems with orbital separations drawn from the T14 model.

2.4 Eccentricity distribution variation

In this section we compare the effect of assuming circular orbits against eccentricities drawn from a thermal distribution. We generate the parent distributions

by drawing all the orbit parameters as defined in section 2.1 and run one simulation with the eccentricities drawn from a thermal distribution, while the other simulation is run with eccentricity= 0.

Simulation details:

- Sample size for orbital separations: 10000 unique orbital a_{in} a_{out} pairs.
- Sample size for mass ratios: 100.
- Sample size for (thermal) eccentricities: 100.
- Samples size for inclinations: 100.
- Total number of unique orbits for thermal eccentricity simulation: 10^{10}.

The mass ratio distribution plots produced in the following simulations are similar to figures 20 and 24, so we do not repeat them.

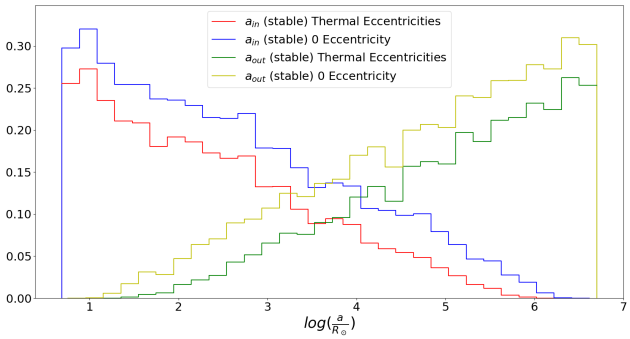

Figure 25: Histogram showing variation in orbital separation distributions for thermal eccentricity distribution vs circular orbits for OBin model

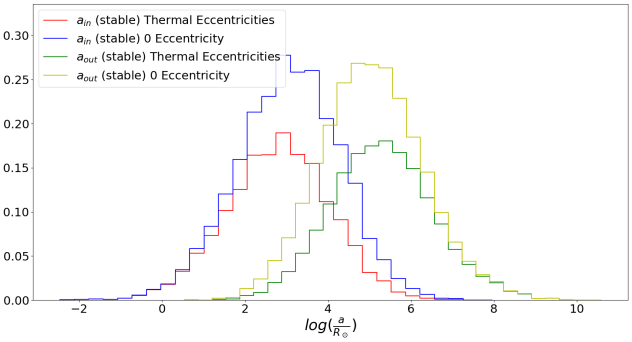

Figure 26: Histogram showing variation in orbital separation distributions for thermal eccentricity distribution vs circular orbits for T14 model

The OBin orbital separation distribution model shows a clear constant offset between the thermal eccentricity and circular orbit distributions (Figure 25), the T14 model (Figure 26) shows a varying offset (in probability for each bin) between the distributions with a larger offset in the moderate orbital separation region. The 0 eccentricity configuration produces more stable systems. The distributions produced by the simulation are similar to the distributions produced in the T14 and OBin general results (Figures 21 and 17), with the 0 eccentricity and thermal eccentricity curves respectively corresponding to the parent and stable distributions. Both the models conclude: the circular orbit distribution is more stable at all orbital separations. The circular distribution case also allows lower ratios of $\frac{a_{out}}{a_{in}}$ since the lowest stable a_{out} and largest stable a_{in} are respectively lower and larger than in the thermal eccentricity case.

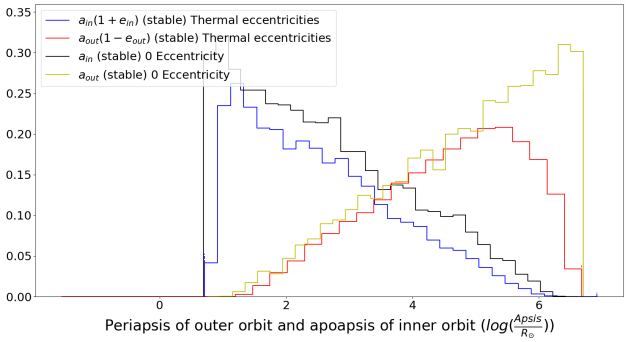

Figure 27: Histograms showing the peripasis distance of the triple star and apoapsis distance of the inner binary for orbital separations drawn from the OBin model (comparison between 0 eccentricity and thermal eccentricity assumptions)

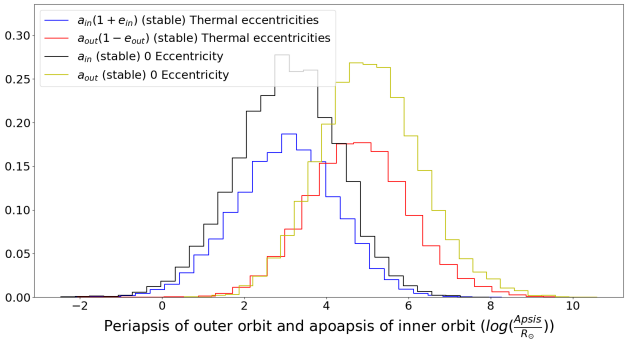

Figure 28: Histograms showing the peripasis distance of the triple star and apoapsis distance of the inner binary for orbital separations drawn from the T14 model (comparison between 0 eccentricity and thermal eccentricity assumptions)

Figures 27 and 28 show the plots comparing the periapsis of a_{out} and apoapsis of a_{in} for the thermal eccentricity and 0 eccentricity cases. We plot the apoapsis distance of the inner orbit and the periapsis distance of the outer orbit as the distance between these distributions indicate the distances at which the inner

binary and triple star are likely to interact. Overall we find the distribution with thermal eccentricities to be closer together in both the OBin and T14 models.

2.5 Outer Mass ratio distribution variation

We run simulations to check the difference between the mass distributions: uniform and $\frac{1}{q}$, figure 29 shows stable and parent histograms of 1000 randomly drawn mass ratios from each distribution.

Simulation details:

- Sample size for orbital separations: 20000 unique orbital a_{in} a_{out} pairs.
- Sample size for mass ratios: 1000.
- Sample size for eccentricities: 100.
- Samples size for inclinations: 100.
- Total number of unique orbits : 2×10^{11}.

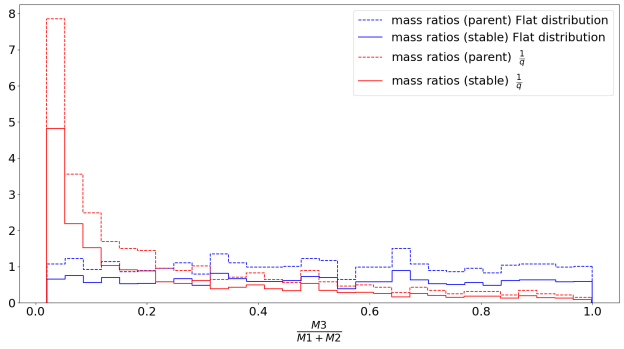

Figure 29: Histogram showing parent and stable mass ratios for flat and $\frac{1}{q}$ distributions.

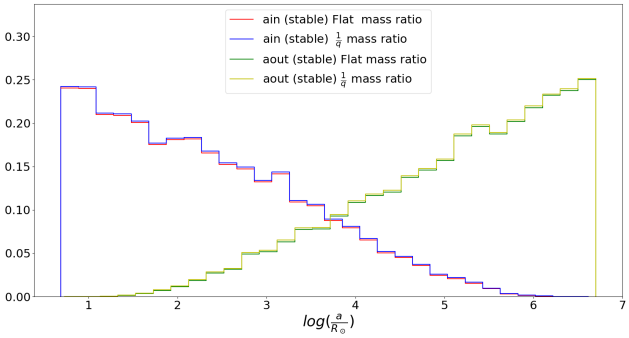

Figure 30: Histogram showing stable orbital separations (OBin model) for flat and $\frac{1}{q}$ mass ratio distributions

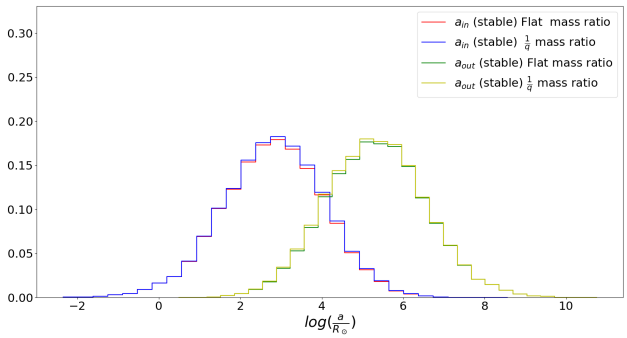

Figure 31: Histogram showing stable orbital separations (T14 model) for flat and $\frac{1}{q}$ mass ratio distributions

Figures 30 and 31 respectively show orbital separation distributions for the OBin and T14 models. For both models we observe an insignificant offset between the flat and $\frac{1}{q}$ mass distributions. The $\frac{1}{q}$ model is found to be marginally more stable for both orbital separation distributions. The stability plot over periapsis distance is not presented here, however a similar trend is observed for both OBin and T14 models, the offset between the two mass ratio distributions is small and the $\frac{1}{q}$ distribution is more stable. No bias can be seen in the uniform mass ratio distribution, however the $\frac{1}{q}$ stable distribution exhibits a bias towards low mass ratios (29).

3 Quadruples

We extend our hierarchical structure by adding a fourth star to the system in the configuration shown in figure 32.

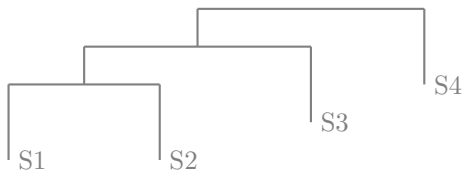

Figure 32: Quadruple hierarchical structure

We redefine (and add) parameters for our quadruple simulation:

- a_{in} : Orbital separation of the two stars within the inner binary (S1 & S2).
- a_{mid} : Orbital separation between the COM of the inner binary and the outer orbiting star (S3).
- a_{out} : Orbital separation between the COM of the inner triple and the outer orbiting star (S4).
- e_{mid} : Eccentricity of S3 orbit around COM of inner binary (S1,S2)

- e_{out} : Eccentricity of S4 orbit around COM of inner triple (S3,(S2,S1)).
- Inclination (inner) : Inclination of orbital plane of S3 with respect to orbital plane of inner binary (S1,S2).
- Inclination (outer) : Inclination of orbital plane of S4 with respect to orbital plane of inner triple (S3,(S2,S1)).
- q_{in} (Inner mass ratio) : $\frac{M2}{M1}$.
- q_{mid} (Middle mass ratio) : $\frac{M3}{M1+M2}$.
- q_{out} (Outer mass ratio) : $\frac{M4}{(M1+M2)+M3}$.

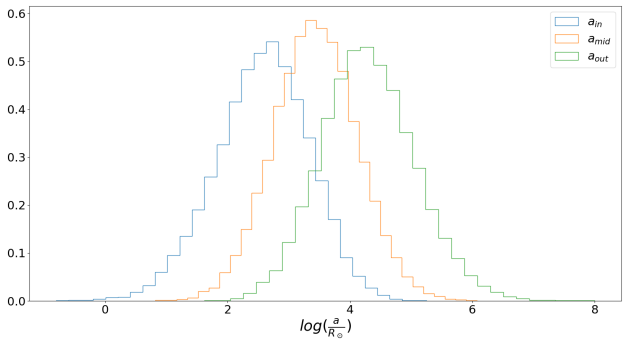

Figure 33: Histogram showing parent distributions of orbital separations for a quadruple system drawn from the T14 distribution

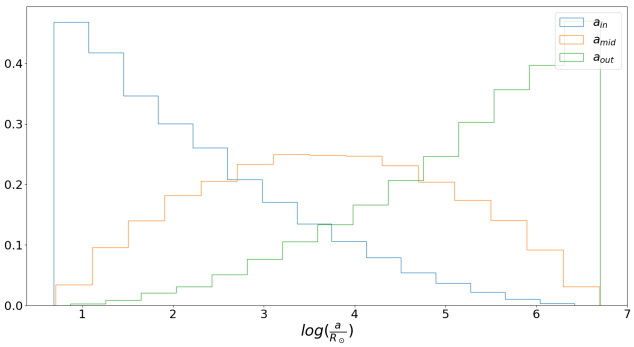

Figure 34: Histogram showing parent distributions of orbital separations for a quadruple system drawn from the OBin distribution

We draw orbital separations from the same distributions as the triple simulations; OBin and T14. We sort the drawn orbital separations into the three variables; a_{in}, a_{mid} and a_{out}. We sort the orbital separations using the following algorithm:

- Draw $3\times N$ (where N is a positive integer) orbital separations from distribution (OBin or T14).
- Select the first three elements of the array containing drawn orbital separations. Sort these into increasing order, append the smallest, middle and largest elements of sorted (section of) array respectively into a_{in}, a_{mid} and a_{out} lists.
- Delete first 3 elements of array containing drawn orbital separations.
- Repeat previous steps until array containing drawn orbital separations is empty.

Figures 33 and 34 show the sorted orbital distributions for T14 and OBin respectively. Note the high peak in a_{mid} in figure 43 is due to the shape of the original normal distribution (Figure 4). The middle value in each orbital configuration is assigned to a_{mid} and the normal distribution is symmetric and peaks at the mean value in the middle of the orbital separation range. Note similarly to algorithm 3 for the triple sorting case, we confirm the lack of biases within the quadruple sorting algorithm in figures 35 and 36.

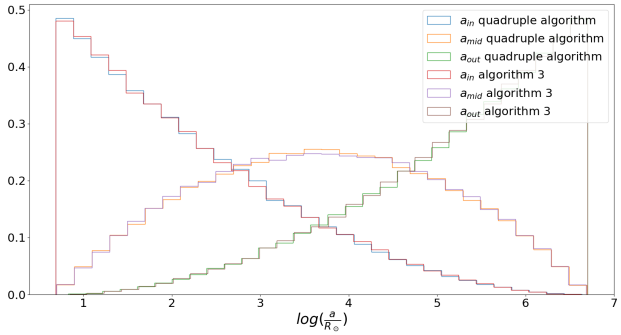

Figure 35: Histograms showing parent distributions of orbital separations for a quadruple systems drawn from the OBin distribution using the standard quadruple algorithm and a version (adjusted for quadruples) of algorithm 3 from section 2.1.2.

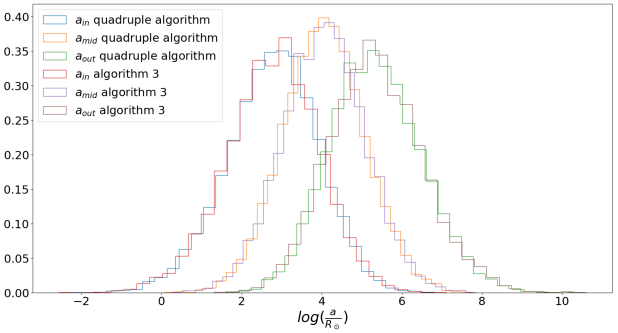

Figure 36: Histograms showing parent distributions of orbital separations for a quadruple systems drawn from the T14 distribution using the standard quadruple algorithm and a version (adjusted for quadruples) of algorithm 3 from section 2.1.2.

$$M1 \times \begin{bmatrix} q_{in_1} & q_{in_2} & \dots & q_{in_N} \end{bmatrix} = \begin{bmatrix} M2_1 & M2_2 & \dots & M2_N \end{bmatrix}$$
$$\begin{bmatrix} M2_1 & M2_2 & \dots & M2_N \end{bmatrix} \times \begin{bmatrix} q_{mid_1} & q_{mid_2} & \dots & q_{mid_N} \end{bmatrix} = \begin{bmatrix} M3_1 & M3_2 & \dots & M3_N \end{bmatrix}$$

3.1 Mass Ratios

The mass ratio array for quadruples is more complicated than in the triple case.

- Initially we consider the triple shown in figure 2 and draw mass ratios for the triple ((S1,S2),S3) (q_{mid}) from a uniform distribution in the same way.
- We draw mass ratios for the inner binary (q_{in}) from a uniform distribution.
- We assume the mass of the primary star in the inner binary is $2M_\odot$, using this assumption and the array of q_{in} we calculate the corresponding array of $M2$ (Since $q_{in} = \frac{M2}{M1}$).
- Since we have arrays of $M2$ and q_{mid}, we are able to calculate an array of corresponding $M3$ values (Since $M1$ is constant).
- For every q_{in} (directly drawn flat distribution) we generate a unique $M2$ which is multiplied by a corresponding q_{mid} element (directly drawn flat distribution) to generate a corresponding $M3$.
- Assuming the mass ratio (q_{out}) of the "triple" configuration (S4,(S3,(S1,S2))) is a uniform distribution, we set the lower bound of the distribution as ($\frac{0.08}{M1+M2+M3}$), the numerator $0.08M_\odot$ corresponds to the minimum mass we assume a star can have.
- We generate a unique q_{out} distribution for every value of q_{mid} (and corresponding $M2$ and $M3$)

Defining the q_{in} and q_{mid}, we set the lower bounds of the flat distributions as being $\frac{0.08M_\odot}{M}$ where M is the maximum "total inner" mass, for the q_{in} distribution $M = 2M_\odot$, whereas for the q_{mid} distribution $M = 4M_\odot$.
In this method of calculating mass ratios, we recursively apply the centre of mass approximation to each layer of the hierarchical structure.

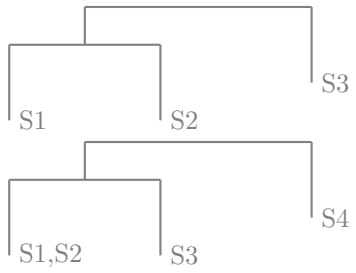

Figure 37: Recursive application of centre of mass approximation to hierarchical structure

3.2 Algorithm

The algorithm for the quadruple system is an extension of the triple system. We use the same initial algorithm to check stability in the triple ((S1,S2),S3) system. If the inner triple is stable, we check the stability of the quadruple by applying the centre of mass approximation to the S1 & S2 system and treat the quadruple as the triple ((S3,(S1,S2)),S4). We assume the same distributions for the inclinations and eccentricities of the outer quadruple system as for the triple systems (thermal distribution for eccentricities, uniform distribution in cosine of inclinations). Algorithm 2 is given in the appendix, the structure of the code remains the same (as the triple case) with added layers to the nested loop to check the quadruple stability. Key differences between the triple and quadruple algorithms are:

- In the triple algorithm, within the innermost for loop, we increment all the counters by "Num", in the quadruple algorithm we increment by "Num_Quad×Num". In the quadruple code "Num" refers to the number of stable triple systems with a unique inner a_ratio, eccentricity and primary mass ratio. For any nonzero number of stable triple systems (found by counting the number of stable inclinations similarly to algorithm 1 for triples) the number of stable quadruple systems (checking across eccentricity, mass ratios and inclinations) are found once (Num_Quad) and multiplied by the number of stable triple systems (Num) to avoid repeats and reduce computational cost (this is used to increment counter variables for all parameters).
- In the triple algorithm the mass ratios were provided in a 1 dimensional array, so a corresponding 1D counter array was created and returned. In the quadruple algorithm mass ratios are given in a 2D array. In section 3.1 we find each value of

q_{mid} generates a unique distribution in q_{out}, the q_ratio array is produced in the format shown below. The first row in the array consists of the q_{mid} distribution (size N). Each column in the array from the second element down consists of the unique q_{out} distribution (size M) generated from the q_{mid} distribution in the first row of that column. For each q_{mid} element (used to check triple stability), we have to iterate over all the elements in that column (from the second row on-wards) as q_{out} when checking quadruple stability. The corresponding mass ratio counter array has the same shape as the mass ratio array (N × M) and contains the counter variables in the same positions as the original mass ratios.

$$\begin{bmatrix} q_{mid_1} & q_{mid_2} & q_{mid_3} & \cdots & q_{mid_N} \\ q_{out_{11}} & q_{out_{12}} & q_{out_{13}} & \cdots & q_{out_{14}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ q_{out_{M1}} & q_{out_{M2}} & q_{out_{M3}} & \cdots & q_{out_{MN}} \end{bmatrix}$$

- The quadruple algorithm initialises and returns counter arrays and variables for both primary and secondary eccentricities.
- Another parameter passed into the fortran library is $a\_ratio2 = \frac{a_{out}}{a_{mid}}$ used to check the quadruple stability.

The arrays returned by the FORTRAN processes are handled in a similar way by the python script as for the triples (section 2.2.2).

3.3 General results

The following subsections contain results showing the stable and parent distributions for quadruples across; Inner binary orbital separation (a_{in}), triple orbital separation (a_{mid}) and quadruple orbital separation (a_{out}). We also plot the stability of the orbital parameters; eccentricities and mass ratios of triple and quadruple orbits.

3.3.1 OBin

Simulation details:

- Sample size for orbital separations: 4000 unique orbital a_{in} a_{out} pairs.
- Sample size for q_{mid} mass ratios: 100.
- Sample size for q_{out} mass ratios: 100.
- Sample size for eccentricities: 100.
- Samples size for inclinations: 100.
- Total number of unique orbits : 4×10^{11}.

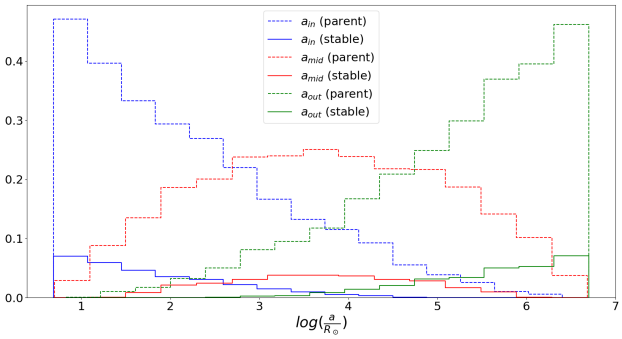

Figure 38: Histogram showing parent and stable orbital separations for quadruples drawn from the OBin model

Figure 38 shows the stable and parent distributions for orbital separations picked from the OBin model (we impose the same cutoffs on the OBin distribution: $[5, 5 \times 10^6]R_\odot$). The parent histograms of a_{in} and a_{out} respectively are skewed towards low and high orbital separations more than the triple case, therefore in the quadruple systems inner binaries are found at lower orbital separations (more often) than in the triple case. The parent distribution of a_{mid} spans the entire orbital separation range peaking in the mid orbital separation region. The parent distributions in the quadruple case show a lot of overlap in the moderate orbital separation region. The a_{mid} distribution overlaps significantly with both a_{in} and a_{out} distributions. The offset between the parent and stable curves is significantly larger than in the triple case, this is expected due to the extra conditions imposed for a stable system.

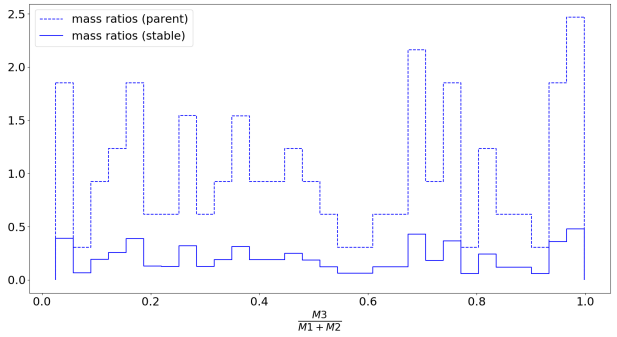

Figure 39: Histogram showing q_{mid} stable and parent distribution for quadruple systems with orbital separations drawn from the OBin distribution

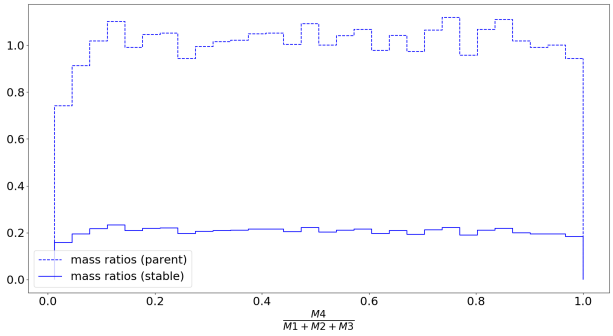

Figure 40: Histogram showing q_{out} stable and parent distribution for quadruple systems with orbital separations drawn from the OBin distribution

Figures 39 and 40 respectively show the stability across the parameters q_{mid} and q_{out}. Similarly to the triple cases we see no bias in stability in mass ratios, both plots show roughly constant offsets between the stable and parent distributions. Both figures show uniform distributions in parent and stable distributions. Figure 40 is affected significantly less by number statistics due to larger sample numbers in q_{out} than q_{mid} due to the process described in section 3.1 (We create unique distributions of q_{out} for each unique element of q_{mid}). Note the very low mass ratio region of figure 40 has a rising non uniform shape as a result of calculating the minimum of the q_{out} distribution based on each unique q_{mid}.

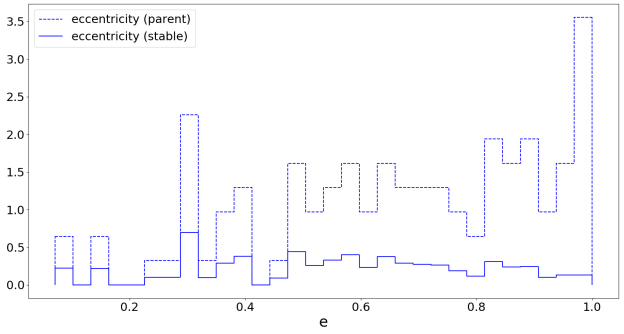

Figure 41: Histogram showing e_{mid} stable and parent distribution for quadruple systems with orbital separations drawn from the OBin distribution

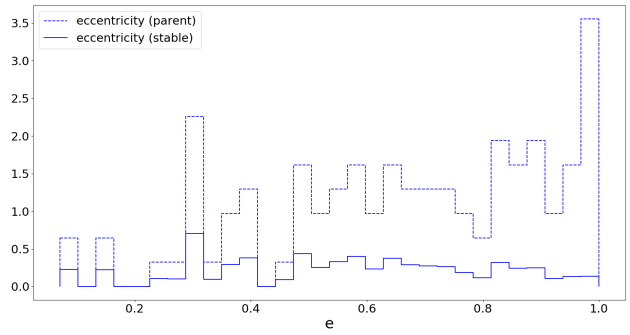

Figure 42: Histogram showing e_{out} stable and parent distribution for quadruple systems with orbital separations drawn from the OBin distribution

Figures 41 and 42 are almost identical, the offset between the parent and stable distributions grow with eccentricity. The number of stable systems found at any eccentricity seems to remain approximately constant. The parent distributions are identical in both figures as a result of reusing the triple eccentricity distribution for the quadruple system (since a thermal eccentricity distribution applies to both of these cases, we reuse the same distribution to reduce number of parameters passed into the FORTRAN script).

3.3.2 T14

- Sample size for orbital separations: 4000 unique orbital a_{in} a_{out} pairs.
- Sample size for q_{mid} mass ratios: 100.
- Sample size for q_{out} mass ratios: 100.
- Sample size for eccentricities: 100.
- Samples size for inclinations: 100.
- Total number of unique orbits : 4×10^{11}.

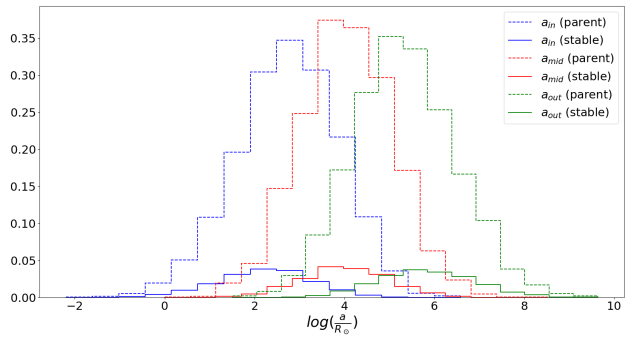

Figure 43: Histogram showing parent and stable orbital separations for quadruples drawn from the T14 model

Figure 43 plots stability across orbital separations drawn from the T14 model. Similarly to the OBin

model (Figure 34) there is significant overlap in the stable distribution between a_{mid} and the other two curves. However there is a much smaller overlap (stable distribution) between the a_{in} and a_{out} curves in the T14 case than in the OBin case. The stable a_{mid} distribution is also significantly more constrained than in the OBin case. Overall the T14 model produces less stable systems than the OBin model (roughly 10% less for quadruples). In both the OBin and T14 quadruple cases we push the a_{in} and a_{out} separations respectively to lower and higher values similarly to the triple cases.

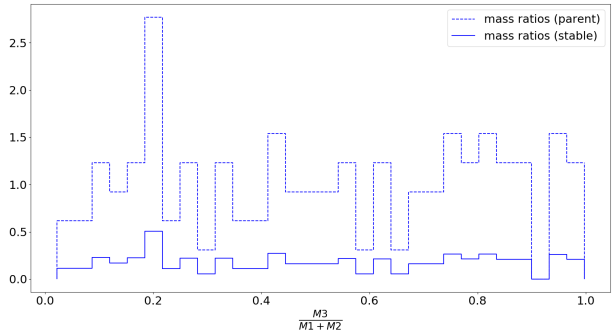

Figure 44: Histogram showing q_{mid} stable and parent distribution for quadruple systems with orbital separations drawn from the T14 distribution

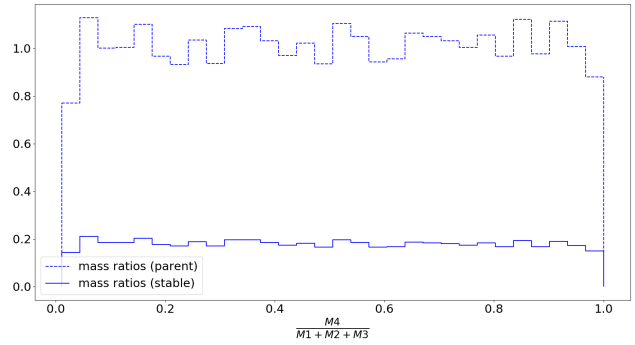

Figure 45: Histogram showing q_{out} stable and parent distribution for quadruple systems with orbital separations drawn from the T14 distribution

Figures 44 and 45 show similar uniform stable and parent distributions as the OBin quadruple mass ratio stability plots (Figures 39 and 40).

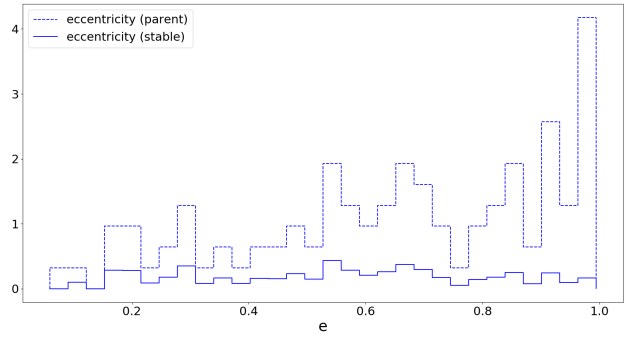

Figure 46: Histogram showing e_{mid} stable and parent distribution for quadruple systems with orbital separations drawn from the T14 distribution

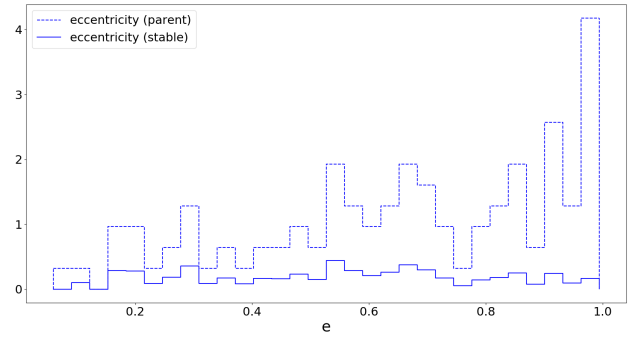

Figure 47: Histogram showing e_{out} stable and parent distribution for quadruple systems with orbital separations drawn from the T14 distribution

Figures 46 and 47 show similar properties to the distributions found for the OBin quadruple eccentricity plots (Figures 41 and 47). Both plots show the number of stable systems found at all eccentricities remains roughly constant and proportionally systems at lower eccentricities are found to be more stable. Both the OBin and T14 systems seem to produce very similar parent and stable distributions for e_{out} and e_{mid}.

4 Summary and discussion

The goal of this project was to investigate the stability of triple and quadruple systems with primaries of $M = 2M_{\odot}$. We investigated the stability of triple and quadruple systems across a range of orbital elements, focusing on any bias or lack of bias in the stable distributions of orbital elements.

In both the OBin and T14 orbital separation plots (Figures 17 and 21) we observe the a_{in} and a_{out} stable distributions respectively shifted to lower and higher orbital separations (compared to parent distributions). We observe a similar pattern for both models in the circular orbit assumption curves (Figures 25 and 26).

We find the following interesting properties about the circular orbit assumption case:

- In the circular orbit assumption we find that since the a_{in} stable distribution is shifted towards lower orbital separations (compared to parent distributions), they are more likely to be interacting inner binary systems (through processes such as mass transfer).
- The OBin model produces more interacting inner binary systems (within the circular orbit assumption) as it densely populates the extreme low orbital separation region (for a_{in}).
- The T14 model produces an a_{in} distribution peaking at higher orbital separation than the OBin model. Due to the larger orbital separation, the inner binary systems are less likely to undergo interactions.
- The T14 a_{out} distribution is situated at lower orbital separation than OBin model, and the a_{in} and a_{out} distributions are found to peak at similar orbital separations (relative to the OBin model). In the case where the inner and outer orbits are found to be close together, the 3 body interactions become important, often leading to dynamical interactions between the triple and inner binary stars.

In the thermal eccentricity case, we see similar orbital separation curves for both the OBin and T14 models (Figures 17 and 21).

- The orbital separation curves are not useful in this case for commenting upon system interactions. Since each system has a non zero eccentricity picked from the thermal distribution, distributions of orbital separation do not necessarily directly impact the probability of interactions within the inner binary.
- The periapsis distance plots (Figures 18 and 22) are useful for discussing the interaction probability of (inner binary) systems since it shows the closest distance of approach between two stars in the inner binary. Lower periapsis distance systems are more likely to interact via mass transfer. We find the OBin model populates the very low periapsis region (outside the shaded regions in figures 18 and 22) more densely than the T14 model. Therefore we expect to find more interacting inner binary systems using the OBin model.
- Since the OBin model is truncated and the T14 model is not, the T14 model produces inner binary systems with orbital separations which are much smaller, however the periapsis distances would require the two stars to overlap (therefore these do not have any physical significance).
- In order to comment on the probability of interactions between the inner binary and the triple star, we look at the periapsis & appopsis plots (Figures 27 and 28). These plots compare the apoapsis distance of inner binary and periapsis distance of the triple (between the circular and thermal eccentricity assumptions). If the appoapsis of the inner binary and the periapsis of the triple star is found it be close, the inner binary is likely to interact with the triple star. For both the T14 and OBin model we find the 0 eccentricity distribution produces systems where the closest approach distance between the triple and inner binary systems are smaller than the thermal eccentricity model (ie more dynamical interactions). In this regime the 3 body dynamics of the system are important. For circular orbits the Kozai Lidov cycles only occur within a specific range of inclinations ($39° - 141°$), whereas in the thermal eccentricity model the cycles are allowed in the whole inclination range.
- In the case of the triples, we start with distinct distributions in orbital separations (OBin & T14), however the shapes of the stable curves are similar (Figures 17 and 21). While the OBin model and the T14 model populate different regions of the orbital separation plot, similar patterns of high stability in the extremely low (a_{in}) and high (a_{out}) orbital separations is observed.
- Similarity between the OBin and T14 models is also found in the periapsis distance plots (Figures 18 and 22), the stable distributions found for both the OBin and T14 model are extremely similar in shape and periapsis distance range. Due to the similarity in periapsis distance (in the non interacting region), we conclude we would see similar numbers of non interacting systems at any given periapsis distance produced by the OBin and T14 distributions.

For the quadruple simulations, we find the following interesting properties:

- In the OBin model, we find more systems are pushed to extremely low a_{in} and extremely high a_{out} than in the triple case (Figures 17 and 34), as a result we produce more interacting inner binary systems. However the quaternary star is pushed to higher orbital separations reducing the probability of interaction with the inner systems.
- Note even though we do not show the graphs of $a(1-e)$ for the quadruple systems, from the

triple simulations we found the $a(1-e)$ curves to be similar to the orbital separation distributions shifted to lower orbital radii (due to $a(1-e) \leq a$).

- The tertiary star orbital separation distribution (a_{mid}) is found to have a "semi-circular" (stable) distribution across the whole range of orbital separations (OBin model), overlapping significantly with both the a_{in} and a_{out} distributions (Figure 34).
- In the T14 model all three orbital separations (stable) are constrained to a small region (Figure 43), resulting in significant overlap between the distributions (proportionally compared to the size of the stable distributions).
- Additionally the three stable distributions produced from the T14 model have peaks close together. Due to this the T14 quadruple model produces systems where the 4 body interactions become important.

The parent eccentricity distribution for both the triple and quadruple case is thermal. We find the following interesting features in the eccentricity distributions:

- Over both the triple and quadruple simulations we find lower eccentricity systems to be more stable (Figures 19, 23, 41 and 46).
- The stable eccentricity distribution for the triple system is found to grow in the low eccentricity region and remain uniform in the high eccentricity region (Figures 19 and23). However the quadruple stable eccentricity distribution is found to be uniform across the entire range (Figures 41 and 46).
- Despite the parent distribution for the quadruples being a thermal distribution (which favours high eccentricities), in the stable distribution all eccentricities seem equally probable (Figures 41 and 46). Note proportionally we still find the low eccentricity to be more stable due to the lower probability of the parent distribution in the low eccentricity region.

Comparing the orbital separation distributions produced by the circular orbit assumption and thermal eccentricity distribution, we find thermal eccentricities shifts the (stable) a_{in} and a_{out} curves (Figures 25 and 26) respectively towards lower and higher orbital separations. Since $a(1-e) \leq a$, the thermal eccentricity distribution shifts the closest approach distance (in the circular case this is constant) to lower distances, increasing the probability of interactions in the inner binary.

We find no significant difference (in terms of stability across orbital separations) between the flat and $\frac{1}{q}$ mass distributions (Figures 30 and 31). However we find the $\frac{1}{q}$ distribution produces significantly more stable systems at lower mass ratios than the uniform distribution. The $\frac{1}{q}$ distribution produces a (insignificantly) larger number of stable systems compared to the flat distribution. This could be due to the weak dependence of the stability criteria (Equation 1) on mass ratio.

References

Gaspard Duchêne and Adam Kraus. Stellar multiplicity. *Annual Review of Astronomy and Astrophysics*, 51(1):269–310, 2013. doi: 10.1146/annurev-astro-081710-102602.

D. C. Heggie. Binary evolution in stellar dynamics. *MNRAS*, 173:729–787, December 1975. doi: 10.1093/mnras/173.3.729.

M. B. N. Kouwenhoven, A. G. A. Brown, S. F. Portegies Zwart, and L. Kaper. The primordial binary population. II.. Recovering the binary population for intermediate mass stars in Scorpius OB2. *AAP*, 474:77–104, October 2007. doi: 10.1051/0004-6361:20077719.

R. A. Mardling and S. J. Aarseth. Tidal interactions in star cluster simulations. *MNRAS*, 321:398–420, March 2001. doi: 10.1046/j.1365-8711.2001.03974.x.

Maxwell Moe and Rosanne Di Stefano. Mind your ps and qs: The interrelation between period (p) and mass-ratio (q) distributions of binary stars. *The Astrophysical Journal Supplement Series*, 230, 06 2016. doi: 10.3847/1538-4365/aa6fb6.

Andrei Tokovinin. From binaries to multiples. II. hierachial multiplicity of F and G dwarfs. *The Astronomical Journal*, 147(4):87, mar 2014. doi: 10.1088/0004-6256/147/4/87.

Virginia Trimble. The distributions of binary system mass ratios - A less biased sample. *MNRAS*, 242: 79–87, Jan 1990. doi: 10.1093/mnras/242.1.79.

5 Appendix

Algorithm 2 FORTRAN library (Quadruple simulation)

procedure STABLE(a_ratio,eccentricity,inclinations,q_ratio)
    Types: a_ratio: float, eccentricity: float, inclinations: array, q_ratio: float
    inclination_limit = $\frac{\pi}{0.3}\left(1-\frac{a\_ratio}{\left(\frac{2.8}{1+eccentricity}\right)\left(\frac{(1+q\_ratio)(1+eccentricity)}{\sqrt{1-eccentricity}}\right)^{\frac{2}{5}}}\right)$
    `stable_sys = length(where(inclinations>inclination_limit))`
    `Return stable_sys`
procedure SIMULATION(a_ratio_inner,a_ratio_out,eccentricities,inclinations,q_ratios)
    Types: a_ratio_inner: float, a_ratio_outer: float, eccentricities: array, inclinations: array, q_ratios: array
    primary_e_counter_array = array([0]×length(eccentricities))
    secondary_e_counter_array = array([0]×length(eccentricities))
    q_ratio_counter_array= array([0]×shape(q_ratios))
    a_counter = 0
    for q_primary_index in range(length(q_ratios[1, :])) **do**
        mass_slice = q_ratios[:,q_primary_index]
        q_primary = mass_slice[0]
        `q_counter_primary=`0
        for eccentricity_primary_index in range(length(eccentricities)) **do**
            eccentricity_primary = eccentricities[eccentricity_primary_index]
            e_primary_counter = 0
            `Num=Stable(a_ratio_inner,eccentricity_primary,inclinations,q_primary)`
            if Num $\neq$ 0 **then**
                q_secondary_index = 1
                for q_secondary in mass_slice[1 :length(mass_slice)] **do**
                    for eccentricity_secondary_index in range(length(eccentricities)) **do**
                        eccentricity_secondary = eccentricities[eccentricity_secondary_index]
                        Num_Quad = `Stable(a_ratio_out,eccentricity_second,inclinations,q_secondary)`
                        `q_counter_primary += Num_Quad`×`Num`
                        `a_counter += Num_Quad`×`Num`
                        `e_primary_counter += Num_Quad`×`Num`
                        secondary_e_counter_array[eccentricity_secondary_index] += Num_Quad×Num
                        `q_counter_secondary += Num_Quad`×`Num`
                    q_ratio_counter_array[q_secondary_index,q_primary_index] += q_counter_secondary
                    q_secondary_index += 1
            primary_e_counter_array[eccentricity_primary_index] += e_primary_counter
        q_ratio_counter_array[0,q_primary_index] += q_counter_primary
    `Return a_counter, primary_e_counter_array, secondary_e_counter_array, q_counter_array`