Draft version April 15, 2020
Typeset using LaTeX **twocolumn** style in AASTeX63

Bayesian Detection of Invisible Planets

Delaney Dunne,[1] Michael Lindner-D'Addario,[1] and Gabriella Morin[1]

[1] *Department of Physics, McGill University*
3600 Rue Université,
Montréal, QC, H3A 2T8, Canada

(Received April 14, 2020; Revised April 14, 2020; Accepted April 15, 2020)

Submitted to AJ

ABSTRACT

Using Bayesian inference, the detection of the unknown masses of planets and stars was attempted by simulating the trajectory of N known bodies in the presence of m unknown bodies in two dimensions. The invisible planets were then detected by applying a Markov-Chain Monte Carlo (MCMC) algorithm to a posterior distribution for the mass of the unknowns built from the probability that their presence with a given mass gave rise to the trajectory simulated. This probability was estimated from the Gaussian difference between the simulated trajectory and the trajectories obtained with the unknowns at each mass. The program was found to be quite sensitive to unknown masses that caused only small perturbations in the trajectories of the known masses, but gave incorrect results for prior distributions which resulted in instability. Provided with the Sun and a few other known planets from the solar system, it was found to be possible to get a good approximation on the mass of a planet in the system made "invisible" for the purpose of the simulation. Given knowledge of the planets in the solar systems, and reasonable constraints on the prior distribution, the mass of the sun was determined to 0.3% of the actual.

Keywords: Bayesian statistics, solar system dynamics, N-body simulations, Markov-Chain Monte Carlo

1. INTRODUCTION

Unlike stars, planets do not emit light. Their presence is consequently harder to detect. If their orbits intersect with the line of sight of a telescope and their host star, the red/blue shift of the light from the star will testify of the presence of this non-luminous object. But not all orbits allow for this detection method. Given that most planets have stable orbits around their host star and that the physics of the gravitational forces between these non-relativistic objects of large masses is well understood, we applied computational tools to model the trajectories of planets around a star. Given that the number and mass of bodies in a N body simulation greatly affects the trajectories obtained, we hoped to use this to determine the credibility region with which the presence of m additional bodies and their masses could be confirmed.

Bayesian statistics are of great use in this type of problem since they allow for starting from measured (in the present case, simulated) data and inferring the probability that an assumption or a model is valid given the data. Provided with an $N + m$ body simulation and various possible masses for the m unknown planets or stars, the difference could be measured between the known trajectories of the known objects and their trajectories in the presence of the m unknown bodies. This defines a likelihood probability distribution for the mass of each of the unknown bodies. Given a prior knowledge that the mass may be zero if the assumed unknown bodies turns out not to be part of the system under consideration, but cannot be negative, a uniform prior distribution may be defined for all values of mass greater or equal to zero. Bayes theorem then states that for a simulated trajectory $\vec{t}$, the m unknown masses $\{m_i\}$ can be inferred as,

$$p(\{m_i\}|\vec{t}) = \frac{p(\vec{t}|\{m_i\})p(\{m_i\})}{p(\vec{t})}, \tag{1}$$

where $p(\vec{t})$ does not depend on the masses and can consequently be accounted for by normalizing the probabilities. The probability of the m unknown bodies having masses defined by the set $\{m_i\}$ given the measured

trajectory is determined with the posterior, $p(\{m_i\}|\vec{t})$. This posterior distribution can be efficiently sampled using the random biased sampling method of the Markov-Chain Monte Carlo (MCMC) algorithm.

In addition, many possible scenarios exist in which it is known that a body exists, and a rough estimate of its mass is attainable (for example, from using the stellar mass-luminosity relation when an approximate luminosity and a distance are known), but additional constraints on the mass of the body are required. In these cases, harsher prior distributions can be adopted, and the same method applied.

In this project, we applied Bayesian statistics and a Markov-Chain Monte Carlo algorithm to the determination of the mass of an unknown body, given its position and the masses and trajectories of other bodies interacting with it gravitationally. All programming was done using Python. All quantities of time, length, and mass hereafter are respectively defined in units of days, astronomical units (AU), and solar masses $M_\odot$.

2. METHODS

2.1. *N-body Simulation*

In order to model the motion of astrophysical objects, based only on initial position and velocity conditions and variable masses, a gravitational simulation was created. This simulation modeled the motion of N astrophysical bodies as the solution to a system of $4N$ first-order differential equations, where the i^{th} equation is given by,

$$m_i\vec{a}_i = \sum_{j=1,j\neq i}^{N} \frac{Gm_im_j(\vec{r_i}-\vec{r_j})}{||\vec{r_i}-\vec{r_j}||^3}. \tag{2}$$

At each step in the simulation, the net acceleration $\vec{a}$ of each body was calculated using Eq. (2). It was assumed the body experienced this acceleration for the entire duration dt of the step, and the final position and velocity of the body was determined based on this constant acceleration,

$$\begin{aligned} \vec{x}_{final} &= \vec{x}_{init} + \vec{v}_{init}\ dt + \frac{1}{2}\ \vec{a}\ dt^2 \\ \vec{v}_{final} &= \vec{v}_{init} + \vec{a}\ dt \\ t_{final} &= t_{init} + dt. \end{aligned} \tag{3}$$

Here, $\vec{x}_{init}$ and $\vec{v}_{init}$ are respectively the position and velocity of the body before the timestep, and $\vec{x}_{final}$ and $\vec{v}_{final}$ are its position and velocity after. In order to keep the problem tractable, all bodies were confined to the (x, y) plane. Cartesian coordinates were used. After each step in the simulation, the position vector for each body was printed to a file, for further analysis.

Finally, boundary conditions were imposed on the simulation - any body reaching a distance of 1 000 AU from the center of mass of the system would collide completely inelastically with a figurative boundary wall. This way, when the N-body system was unstable, bodies reaching large distances (possibly resulting in NaN values) would not cause instabilities. Periodic boundary conditions were considered, but the interpolation required to construct the surrogate model made large jumps in position risky.

To visually confirm bodies were moving gravitationally, the position data output by the simulation was animated using Python's `pyglet` package. Real position and velocity data from our solar system were used as initial conditions, as the solar system displays predictable orbital behaviour and therefore the values generated by the simulation could be compared to observations. Ephemerides for solar-system bodies were taken from NASA's HORIZONS system (Giorgini & JPL Solar System Dynamics Group (2001)).

2.2. *Bayesian Statistics*

To manipulate the position data output by the N-body simulations, the trajectories $\vec{t}$ were formatted into three-dimensional arrays,

$$\vec{t} = T_{ijk}. \tag{4}$$

The three indices correspond to the given time step of the simulation t_i, the position coordinate x_j, and the index of the body N_k, respectively.

The goal is to compare the simulations for systems with arbitrary masses $\{m_i\}$ to a simulation with specific masses $\{\vec{m}_i\}$ and to use Bayesian statistics to infer those specific masses. In order to use Eq. (1), it was necessary to devise a likelihood function. The Gaussian function,

$$P(\vec{t}|\{m_i\}) = \prod_{i,j,k} \frac{1}{\sqrt{2\pi\eta}} \exp\left[\frac{-(\vec{t}-\vec{t}_{\rm real})^2}{2\eta}\right], \tag{5}$$

was chosen. Here, $\vec{t}$ corresponds to the trajectory obtained by running the N-body simulation with an arbitrary set of masses $\{m_i\}$, while $\vec{t}_{\rm real}$ corresponds to that obtained by using the true unknown masses $\{\vec{m}_i\}$ of the given system that is under study. The indices i, j, k refer to the definition of the trajectories in Eq. (4). The parameter η controls the width of the distribution and will assert the degree of precision with which $\{m_i\}$ will have approached $\{\vec{m}_i\}$.

The priors $p(\{m_i\})$ used in Eq. (1) were chosen to be uniform and non-negative. In the event that the parameter η was also chosen to be subject to Bayesian inference, it was also given a uniform, non-negative prior.

A large number of trajectories $\vec{t}$ need to be computed to perform Bayesian inference. Therefore, to reduce computational efforts and to speed up run times, a surrogate model was constructed. For an inference problem with m unknown masses $\{m_i\}$, $i = \{1, 2, ..., m\}$, the model was constructed by conducting the simulations at s evenly spaced values along each of the uniform $\{m_i\}$ priors, resulting in an m dimensional model with s^m points.

In addition to constructing the surrogate model, one final simulation needed to be ran using the true unknown masses $\{\vec{m}_i\}$ in order to use Eq. (5). Once these $s^m + 1$ simulations were completed, no other expensive simulations needed to be conducted. The surrogate model circumvents additional N-body simulations through the use of interpolation. For any set of masses $\{m_i\}$, the corresponding trajectories were determined by linearly interpolating the trajectories of the surrogate model, an operation that is orders of magnitude faster than running a new N-body simulation.

The accuracy of the interpolation scheme as well as its ability to locate the underlying global maxima is seen in Fig. (1). Additionally, the application of this inference to multiple unknown bodies is seen in Fig. (2).

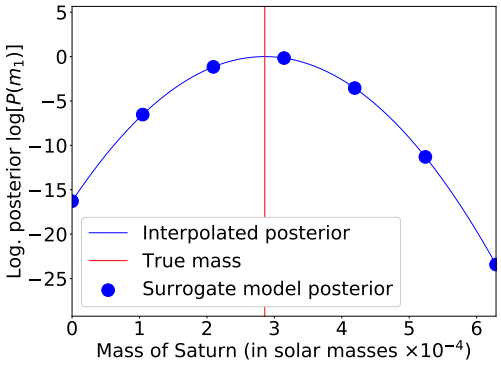

Figure 1. Statistical analysis of Sun-Jupiter-Saturn system. Here, the trajectories of the Sun and of Jupiter are known, and the mass of Saturn is treated as the unknown. The value of η has been set to 1 for simplicity. Ephemerides taken from Giorgini & JPL Solar System Dynamics Group (2001).

2.3. *MCMC Algorithm*

The MCMC algorithm was carried out using the `emcee` package for Python with the posterior distribution defined in the mass space of the unknown bodies.

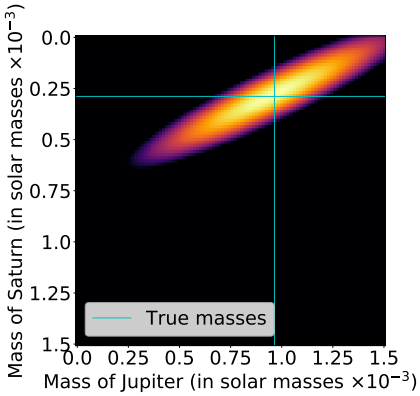

Figure 2. Statistical analysis of Sun-Jupiter-Saturn system. Here, the trajectory of the Sun is known, and the masses of Jupiter and Saturn are treated as the unknowns, respectively. The value of η has been set to 1 for simplicity. Ephemerides taken from Giorgini & JPL Solar System Dynamics Group (2001).

9 surrogates points and 16 walkers were used for all systems to interpolate and sample the distributions. The variable η, which determined the width of the likelihood probability distribution, was either included as a parameter to the MCMC algorithm and then marginalized over, or set to a plausible non-zero positive value and excluded from the list of parameters. In the later case, the most probable mass from the posterior distribution determined by the algorithm was considered informative, but the width of the distribution was no longer associated with the credibility region about this most probable mass since it was dependent on the arbitrary η.

3. RESULTS AND DISCUSSION

3.1. *Toy models*

To test the reliability of the gravitational simulations, we simulated and animated the orbit of Europa, one of Jupiter's moons, to see if this orbit would remain stable around Jupiter while the planet orbited the Sun. The success of this simulation can be seen in Fig. 3. To achieve stability of the orbit, a time step dt as small as 14.4 minutes was required. When using time steps that were larger, Europa was rather observed to "fly away" from Jupiter and so the motion was not stable. This behavior was attributed to the approximation of assuming constant acceleration over a small interval dt and moving the masses accordingly instead of integrating the gravitational force equation. The period of revolution of Europa around Jupiter is 85 hours, and therefore the dt needed to be small in comparison to ensure stable gravitational motion.

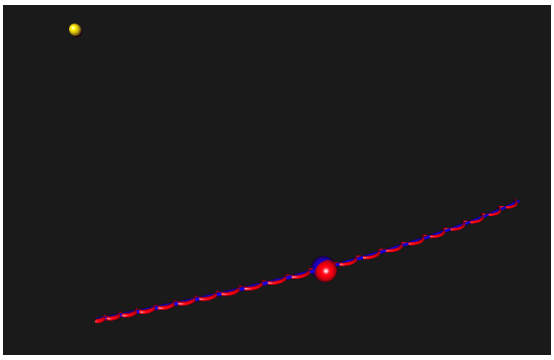

Figure 3. A frame from the `pyglet` simulation of the Sun, Jupiter and Europa, run at a dt value of 0.01. At time steps this small, the simulation was capable of modelling the motion of all three bodies.

As a more quantitative test, the generated trajectories were compared to those given by Kepler's laws of orbital motion,

$$r = \frac{a(1-e^2)}{(1+e\cos(\theta-\theta_0))}, \tag{6}$$

where a is the length of the semi-major axis and e is the eccentricity of the ellipse. The angular position, θ, has a constant velocity $\frac{d\omega}{dt}$, so the position as a function of time can be plotted using this equation. When calculated using Jupiter's period and elliptical parameters and a timestep of 0.5 days, this trajectory agreed with Jupiter's trajectory in a simulated Sun-Jupiter-Saturn system (shown in Fig. (4)) to 2.7%.

A set of toy model systems was created to test the different parts of the project without being too computationally expensive. These were consequently run with a default low number of iterations for both the simulation and the MCMC algorithm, since the goal was only to obtain an idea for the behavior of code and not to reach high levels of accuracy.

The first of these toy models was a system composed of Jupiter orbiting the Sun with no unknown bodies. This system was first studied with the MCMC algorithm by fitting for only the mass of one allowed unknown body. For this, η was set to 1. The result was a distribution for the mass which peaked at zero with some finite width (see Fig. (5)). When both η and the mass of the unknown body were fitted for, the algorithm converged to a much narrower distribution, again peaked at zero, for the mass of the unknown. As can be expected, there was no difference between the known trajectories of the 2 known bodies and their trajectory in the presence of an unknown of zero mass, which had η also converge toward zero. This made the mass posterior distribution tend to a delta function as shown in Fig. (6).

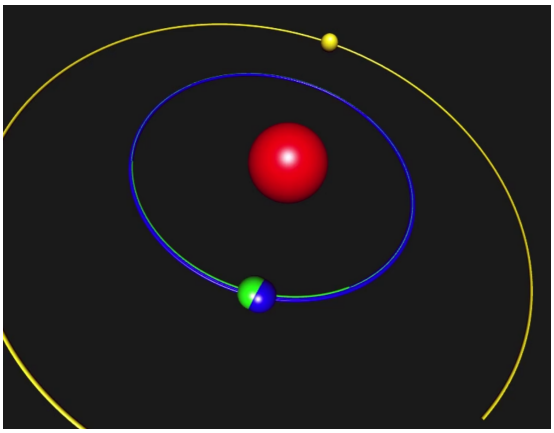

Figure 4. A frame from the `pyglet` animation of the simulated Sun-Jupiter-Saturn system, compared to the analytical Kepler orbit of Jupiter (shown in green). The sun is shown in red, Jupiter's simulated trajectory is blue, and Saturn's is yellow. Masses and distances are scaled up to be more clearly visible, but remain proportional to their original values, and trajectories are traced so the shape of the orbits can be compared. Shown are 20000 iterations of the N-body simulation, at a timestep of 0.5 days. The simulated trajectory of Jupiter here agrees with the analytical version to 2.7%.

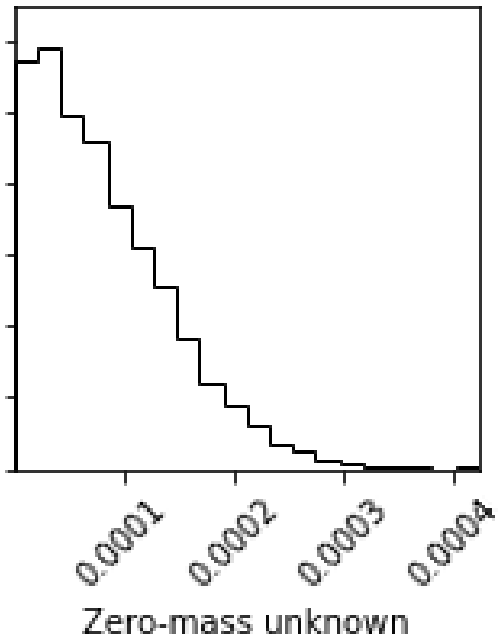

Figure 5. MCMC results for the mass distribution of the unknown body in a system where the trajectory of the Sun and Jupiter are known and no unknown body is present. In attempting the detection of an unknown, its mass probability distribution is expected to peak at zero to reflect that and this results show this. The simulation was run with 20,000 iterations and the MCMC algorithm with 5,000 iterations.

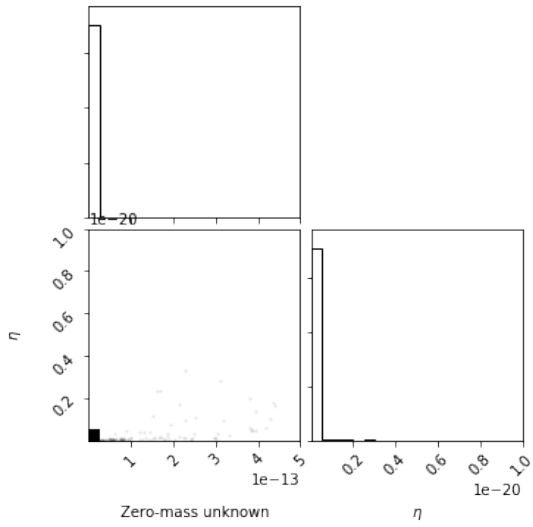

Figure 6. MCMC results for the mass distribution of the unknown body in a system where the trajectory of the Sun and Jupiter are known and no unknown body is present. η was also included as a parameter of the MCMC; it was fixed at one. The simulation was run with 20,000 iterations and the MCMC algorithm with 5,000 iterations.

In the second model, 1 unknown body of non-zero mass was introduced. This unknown body was given a mass, initial position, and initial velocity corresponding to Saturn, with the two known bodies again representing the Sun and Jupiter. η was included in the parameters of the MCMC to obtain an order of magnitude idea for the width of the credibility region achievable. The MCMC for this system resulted in a probability distribution for the mass of the Saturn-like unknown peaked near 2.8035×10^{-4} $M_\odot$ with a 16%-84% credibility region of 2.803×10^{-4} $M_\odot$ to 2.804×10^{-4} $M_\odot$ (seen in Fig. 7). The mass of Saturn used for the simulation was 2.85802×10^{-4}.

Although the actual mass was outside of the credibility region, the low percentage difference between the two peaks of the distribution and the real mass was a pleasant surprise given the number of simplifications and approximations made in the simulation and the likelihood distribution. None of us expected good quantitative agreement from this $N+m$ body system. Furthermore, the initial x and y Cartesian positions of the planets were taken from NASA's HORIZONS systems by simply disregarding the z-component of these planet's position and copying the x and y values. Hence if a planet happened to be close to the Sun in the x-y plane, but had a z-component of position that was much larger, this method of initializing position would modify the distance between that planet and the Sun, thus affecting the quantitative results obtained. However, distances in the z-direction were typically quite small (of $\mathcal{O}(10^{-3})$), and therefore likely did not affect the motion of the system greatly.

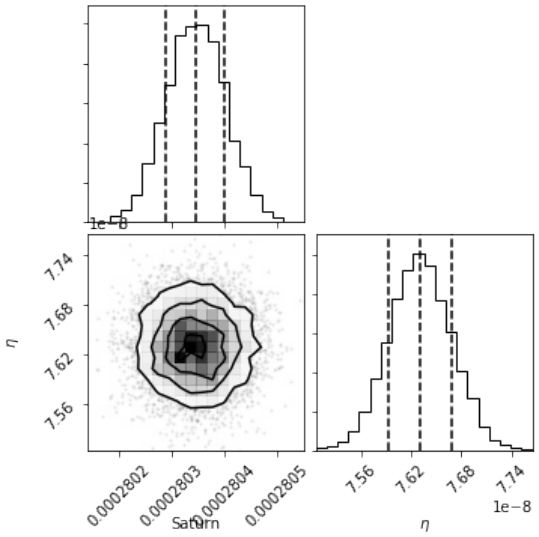

Figure 7. MCMC results for the mass distribution of the unknown body in a system where the trajectory of the Sun and Jupiter are known and one unknown body is present and represented by Saturn. Lines through the histogram represent the 16, 50 and 84% quantiles and η was included in the MCMC parameters. The simulation was run with 20,000 iterations and the MCMC algorithm with 5,000 iterations.

Finally, the inverse system, where Saturn is the only known planet and both Jupiter and the Sun are unknown bodies, was tested. η was fixed at 1 for this system. Due to the additional degree of freedom and the much larger perturbation to the system by making the Sun unknown, the MCMC algorithm converged to a different set of masses for the Sun and Jupiter than these bodies' actual masses. The distribution for the mass of the Sun was peaked near half of its actual mass. The mass of Jupiter was peaked near 0.0525253 $M_\odot$, greater that its true mass of 9.547919×10^{-4} $M_\odot$ (see Fig. (8)).

3.2. *Our solar system*

After studying these toy model systems, we became interested in looking at larger, more realistic systems. We thus looked into modelling all eight planets of our solar system. For this, the Sun was made an unknown body since it was expected that variations in its mass would have the strongest effect on the trajectory of all other known bodies and would thus allow for the simulation and algorithm to more easily converge to a quantitatively reasonable mass. This was not true, as Fig. (9) shows that the MCMC instead converged to a mass of the Sun of roughly 47.5% of its actual mass with very

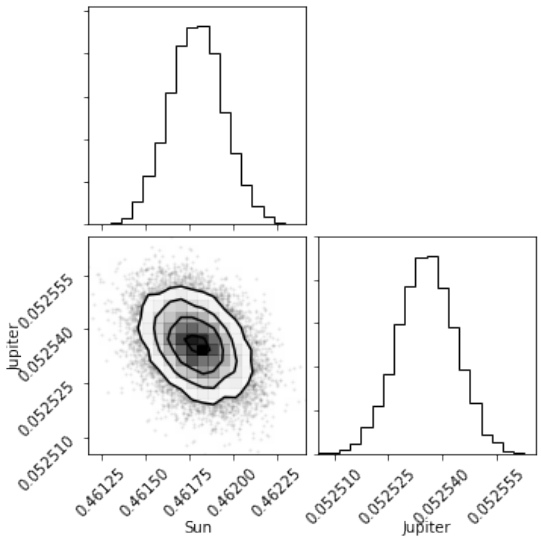

Figure 8. MCMC results for the mass distribution of the unknown bodies in a system where the trajectory of Saturn is known and two unknown bodies are present, namely the Sun and Jupiter. In attempting the detection of these unknowns, η was fixed at one. The simulation was run with 40,000 iterations and the MCMC algorithm with 10,000 iterations.

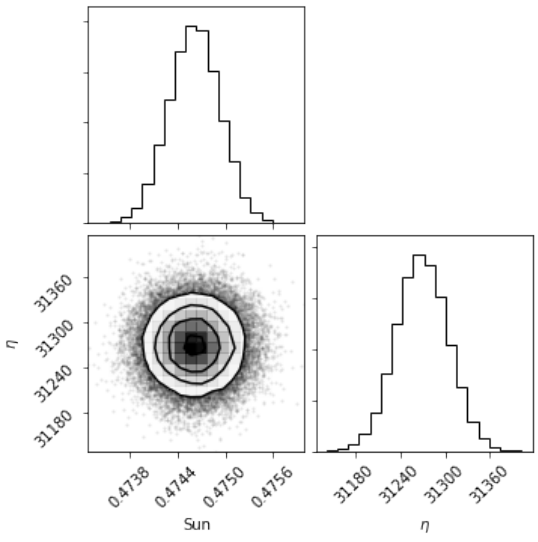

Figure 9. MCMC results for the mass distribution of the unknown body in a system where the trajectories of all planets in the solar system, from Mercury to Neptune, are known, but the Sun is an unknown body. η was included in the parameters to the MCMC algorithm. The simulation was run with 100,000 iterations and the MCMC algorithm with 30,000 iterations.

large η. The algorithm was therefore rerun using a more constraining prior for the mass of the Sun such that it was forced to lie between 50% and 150% of its mass. The initial position of the walkers was also constrained to lie in the range 1.0-1.1 $M_\odot$. Since many methods exist to estimate the mass of stars, it is not unreasonable to assume that in the context of real experiments, these methods could be used to narrow down the prior distribution before running the MCMC algorithm for the determination of the mass. The results of this Bayesian inference with narrow prior is shown in Fig. (10). When reasonable constraints were applied to the prior distribution, the Bayesian determination of the mass of the sun agreed with the sun's actual mass to 0.3%.

Another aspect of interest was to test the limits of detection of this Bayesian inference method and the reliability of the simulation. To test the former, the detection of Pluto, a light and far away body, was performed. The mass distribution for Pluto as determined was extremely narrow and peaked at 6.57279×10^{-9}, in agreement with the actual mass of Pluto passed to the simulation module (see Fig. (11)). The method thus succeeded in detecting an unknown as distant and with as little mass as Pluto.

It should be noted that all of the aforementioned examples only considered up to 2 unknown masses present in the system. The reason for this is due to the linear interpolation scheme implemented with the surrogate model. Although this is a relatively simple procedure

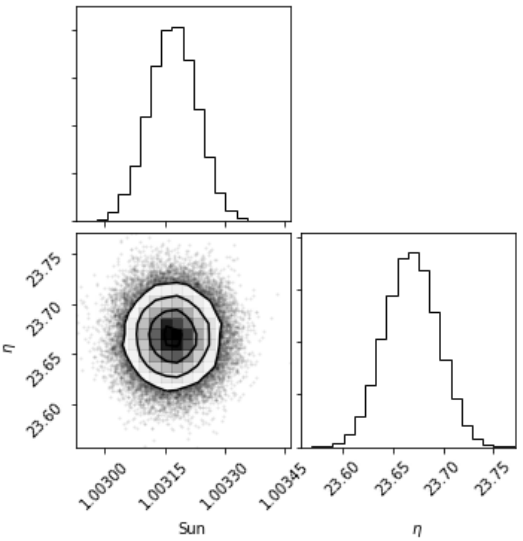

Figure 10. MCMC results for the mass distribution of the unknown body in a system where the trajectories of all planets in the solar system, from Mercury to Neptune, are known, but the Sun is an unknown body. The prior distribution for the mass of the Sun was narrowed down to 0.5-1.5$M_\odot$ for this test. η was included in the parameters to the MCMC algorithm. The simulation was run with 30,000 iterations and the MCMC algorithm with 100,000 iterations.

for 1 or 2 dimensions, interpolation becomes very complicated as the dimensionality increases. It would be of great interest in the future to consider generalized inter-

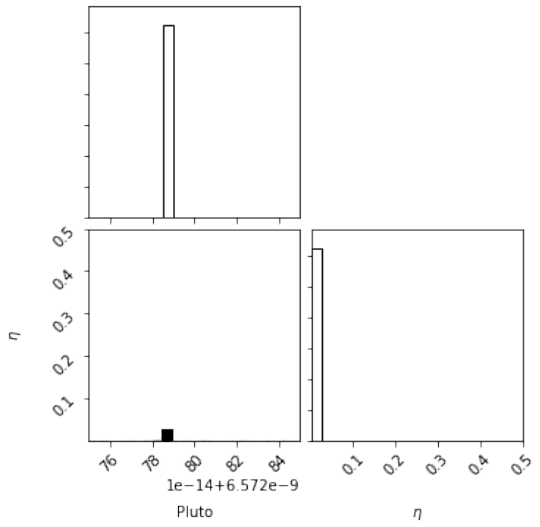

Figure 11. MCMC results for the mass distribution of the unknown body where all planets in the solar system from Mercury to Neptune and the Sun have known trajectories but Pluto is the one unknown body. This was meant to test the detection ability of the Bayesian inference method to small and far objects. η was included as a MCMC parameter. The simulation was run with 50,000 iterations and the MCMC algorithm with 20,000 iterations.

polation schemes to allow for the number of unknown masses to be greater than 2. However, it should also be considered that the accuracy of the surrogate model depends on the density of sampled points, s. As the number of dimensions m increases, the number of points required for the surrogate model grows exponentially as s^m. For the surrogate model to be valid, s^m should be smaller than the number of MCMC iterations. As m grows, this might not remain valid, and therefore at very large dimensions, it would be advantageous to discard the surrogate model methodology.

4. CONCLUSION

The goal of this work was to detect bodies within a solar system by applying Bayesian inference to trajectory differences observed in an N-body simulation with and without additional unknown bodies. To do so, we coded a N-body simulation, from which we extracted the required likelihood probability distribution, in order to apply a MCMC algorithm. Given the simplicity of the simulation and the likelihood function defined, where the acceleration was assumed to be constant over small time step rather than going through the integration of the equations of motion, the differences between trajectories were assumed to correspond to Gaussian errors, and the number of unknown bodies was limited to 2 to allow for simple interpolation between points in mass space where the simulation was run, the quantitative quality obtained for some of the results was very successful.

When working with an unknown of larger mass than the known bodies, such as the Sun, the prior had to be made narrow to allow for the mass probability distribution to peak near the actual mass of the unknown body. However, once the uniform prior was made narrower, the inferred mass of the Sun was 0.3% away from the true value. Such small percentage differences between inferred and known masses of planets were also achieved with systems where there was one planet as the only unknown body. These unknown planets did not need a narrow prior for the accuracy in the inferred mass to be achieved. The method failed in the case of 2 unknown bodies, probably due to the approximations taken which did not allow us to handle such a complex system properly. However, the only test we ran with two unknowns included the sun as an unknown mass, and it is likely the effect of the sun was so large it overpowered that of the smaller planetary mass.

Gravitational motion in general, apart from exceptional cases, is not stable, and therefore large variations in the mass of any single body often resulted in masses escaping the solar system. The program actually responded considerably better to small perturbations in the system's motion (including such small changes as the addition of Pluto to the solar system), where no masses escaped across the entire range of guess masses, than to any prior that resulted in instability. Next steps that could be taken include relaxing some of the simplifications made, and including a third dimension in analysis, for example, to hopefully allow the method to obtain better results even in more complex systems with 2 or more unknown bodies. Finally, the simulation would be considerably more useful if Bayesian statistics could be used to solve for the position and velocity of the unknown masses.

REFERENCES

Giorgini, J., & JPL Solar System Dynamics Group. 2001, On-Line Ephemeris System, http://ssd.jpl.nasa.gov/?horizons