Modeling the Dynamics of Earth-Moon-Sun System Through the 3-Body-Problem

- Averyl Xu
- Damla Temur
- Ian Gu
- Isaiah Liu
- Sambhu Ganesan

“All models are wrong, but some are useful.” - George Box

The N-Body problem investigates how the movement of the celestial bodies can be predicted and how they interact through gravity. We examine this with how the Earth, Moon and Sun move by using 2-body, 3-body and 4-body models using the NBodySimulation function. We compare these models with data from the AstroPosition function, which gives us the positions and velocities of these bodies based on the International Celestial Reference System (ICRS). The results suggest that while making the model more complex improves the system in some ways, also worsens it in other ways as the 4-body model with Jupiter as the fourth body wasn’t always the model with the least errors for some data.

Introduction

The N-Body problem refers to the investigation of the interactions of N-many celestial bodies and how they move. In the case where there are only 2 bodies, formulas like Newton's gravitational rule $F = G\frac{m_1 \times m_2}{r^2}$ were proven to be useful to model the dynamics between the two bodies. For the cases where there are more than 2 bodies, the behaviors of the bodies become more complex and hence require more complex equations.

This problem was attempted to be solved with analytical approaches to model the behaviors of the bodies but to date there are no analytical solutions for the the N-Body problem where $n > 2$. We will compare the data from these 2,3 and 4-body simulations, created with the NBodySimulation function, with the data from the model created using AstroPosition and see how well they align by tracking the positions and velocities of the bodies over time.

Building the 2-Body Model

Setting Up the Initial Conditions

We first need to create our simulation with the NBodyFunction. As this function requires initial values of the masses, velocities and positions of the bodies, we first define their masses and positions here. We can also define how long our simulation will run as 10 years from January 1st, 2000 to January 1st, 2010 using DatePlus and DateRange here.

```
In[1]:= sun = Sun STAR ;
        earth = Earth PLANET ;

In[3]:= sunMass = Sun STAR [ mass ];
        earthMass = Earth PLANET [ mass ];

In[5]:= dates = DateRange[ Sat 1 Jan 2000 00:00:00 UTC ,
          DatePlus[ Sat 1 Jan 2000 00:00:00 UTC , UnitConvert[ 10 yr , "Seconds"]]];
```

There are different reference systems for the positions of the bodies like Sun or Earth-centric systems, the two most popular being the International Celestial Reference System (ICRS) and the Heliocentric System. We will use the International Celestial Reference System. This model will provide a stationary reference point whereas if the model was heliocentric, as the Sun is moving through space, we would get a changing origin. To make calling position data from AstroPosition easier and faster as we used it multiple times, we will defined the positionICRS as a function.

```
In[6]:= positionICRS[body_, date_] := UnitConvert[
    Quantity[AstroPosition[body, {"ICRS", "Date" → date}, "Cartesian"]["Data"],
     "AstronomicalUnit"], "Meters"];
initialSunPositionICRS1 = positionICRS[sun, Sat 1 Jan 2000 00:00:00 UTC];
initialEarthPositionICRS1 = positionICRS[earth, Sat 1 Jan 2000 00:00:00 UTC];
```

We have already defined the positions and the masses of our bodies. Now we will define the velocities. To get the Earth’s and Sun’s velocities as a vector, we will find the difference between the positions of the Earth and the Sun on the initial date (1st January 2000 00:00:00 UTC) and 0.5 minutes later. By dividing it by the change in time (0.5 minutes), we get their velocities in vector form.

```
In[9]:= initialSunPositionICRS2 = positionICRS[sun, Sat 1 Jan 2000 00:00:00 UTC + 0.5 min];
initialEarthPositionICRS2 = positionICRS[earth, Sat 1 Jan 2000 00:00:00 UTC + 0.5 min];
initialSunVelocityICRS =
  (initialSunPositionICRS2 - initialSunPositionICRS1) / UnitConvert[0.5 min, "Seconds"];
initialEarthVelocityICRS = (initialEarthPositionICRS2 - initialEarthPositionICRS1) /
   UnitConvert[0.5 min, "Seconds"];
```

Creating the 2-Body Simulation and Obtaining Data from the Simulation

After setting up the initial conditions, we create the simulation with the NBodySimulation function and run it over 10 years.

```
In[13]:= simulationData2BodyICRS = NBodySimulation["Newtonian", <|sun → <|"Mass" → sunMass,
       "Position" → initialSunPositionICRS1, "Velocity" → initialSunVelocityICRS|>,
     earth → <|"Mass" → earthMass, "Position" → initialEarthPositionICRS1,
       "Velocity" → initialEarthVelocityICRS|>|>, 10 yr, MaxSteps → Infinity];
```

We get the simulation time to make sure it ran for 10 years without the bodies colliding and this will be given in seconds.

```
In[14]:= simulationTime2BodyICRS = simulationData2BodyICRS["SimulationTime"]
```

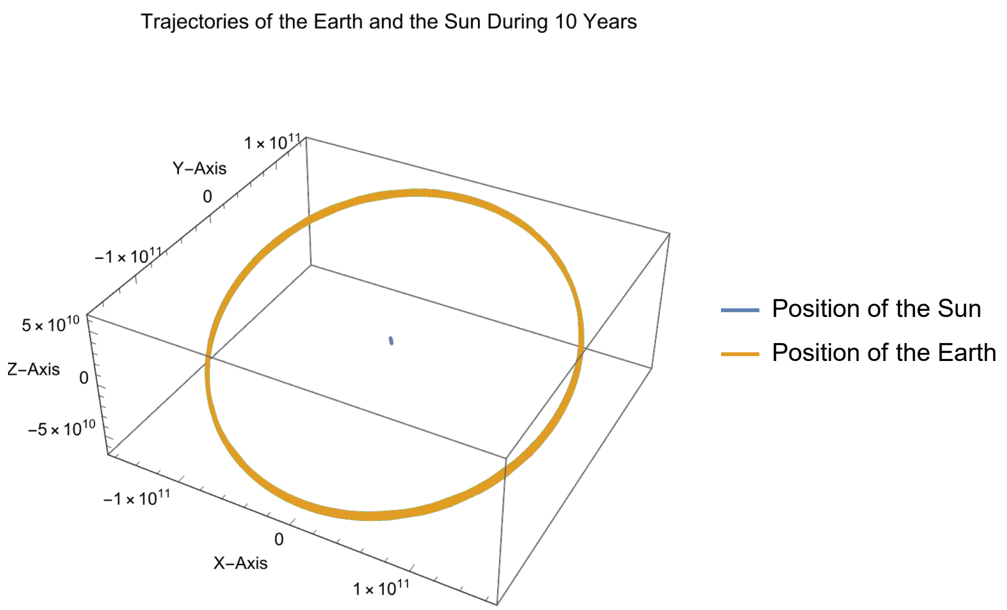
Out[#]=

3.1536×10^8

We plot the positions of the Earth and Sun and it shows the Earth’s nearly circular orbit which seems to hold with the real data in shape.

```
In[24]:= ParametricPlot3D[Evaluate[simulationData2BodyICRS[All, "Position", t]],
  {t, 0, simulationTime2BodyICRS}, AxesLabel → {"X-Axis", "Y-Axis", "Z-Axis"},
  PlotLegends → {"Position of the Sun", "Position of the Earth"},
  PlotLabel → "Trajectories of the Earth and the Sun During 10 Years"]
```

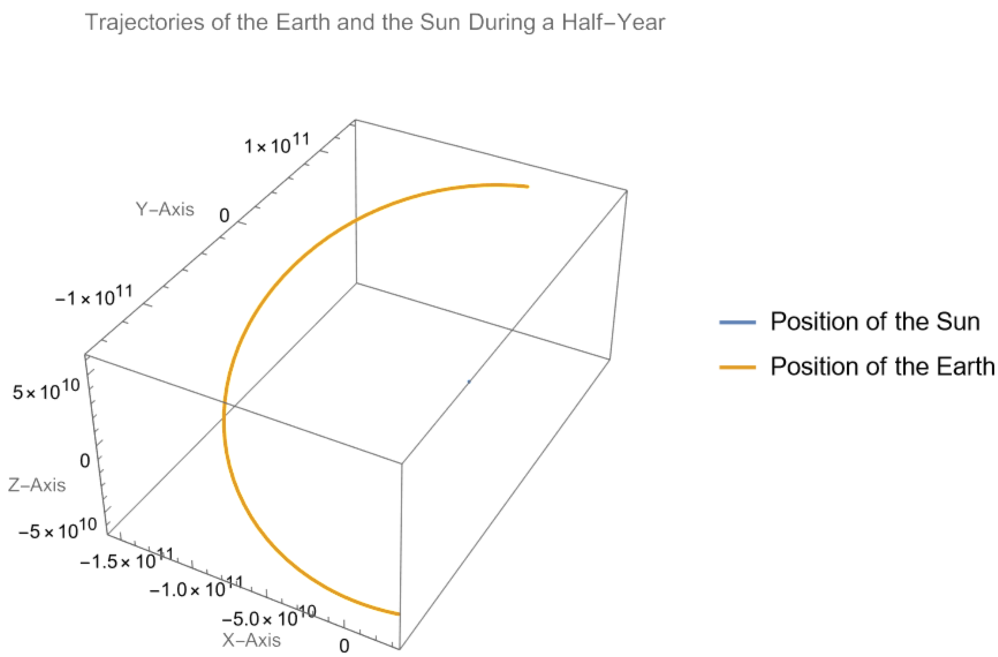
Out[#]=



This is a 10-year plot but the Earth's period around the Sun should be one year. From the plot below, it can be seen that 0.5 years correspond to the displacement of the Earth through half an orbit, hence, suggesting that the simulation showed the Earth's orbit appropriately.

In[25]:=
```
ParametricPlot3D[Evaluate[simulationData2BodyICRS[All, "Position", t]],
 {t, 0, QuantityMagnitude[UnitConvert[ 0.5 yr , "Seconds"]]},
 AxesLabel → {"X-Axis", "Y-Axis", "Z-Axis"},
 PlotLegends → {"Position of the Sun", "Position of the Earth"},
 PlotLabel → "Trajectories of the Earth and the Sun During a Half-Year"]
```

Out[◦]=



We also get the position of each body over time. Again, as this expression is used multiple times, we define it as a function. As our date range was 10 years, getting the position from each second for 10 years takes a long time to run; hence, we got these values each day instead of each second.

In[26]:=
```
simPos2BodyICRS[body_] :=
  Table[Quantity[simulationData2BodyICRS[body, "Position", t], "Meters"],
   {t, 0, simulationTime2BodyICRS, QuantityMagnitude[UnitConvert[ 1 day , "Seconds"]]}];
earthSimPos2BodyICRS = simPos2BodyICRS[earth];
sunSimPos2BodyICRS = simPos2BodyICRS[sun];
```

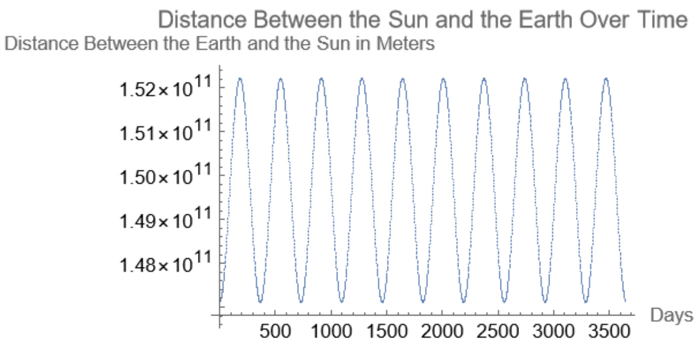
We now get both the distance between the Sun and Earth in vector form and the total distance between them. As we will get the total distances in each model, we define another function here called normDistances that takes the difference between the corresponding elements of each list and as the values are in vector form, it converts them to scalar values with the Norm function.

In[29]:=
```
sunEarthSimDistances2BodyICRS = earthSimPos2BodyICRS - sunSimPos2BodyICRS;
normDifferences[position1_, position2_] :=
  Table[Norm[position1[[i]] - position2[[i]]], {i, 1, Length[position2]}];
normSimDistancesEarthSun2BodyICRS =
  normDifferences[earthSimPos2BodyICRS, sunSimPos2BodyICRS];
```

The plot below shows the total distance between the Earth and the Sun over time as a sinusoidal pattern.

In[56]:=
```
ListPlot[normSimDistancesEarthSun2BodyICRS,
 PlotLabel → "Distance Between the Sun and the Earth Over Time",
 AxesLabel → {"Days", "Distance Between the Earth and the Sun in Meters"}]
```

Out[•]=



We can also get the velocities of each body for each day. The velocities of the bodies are in vector form, to get net velocities, we defined a function called normData which will give the norm of each element that is a vector in the list.

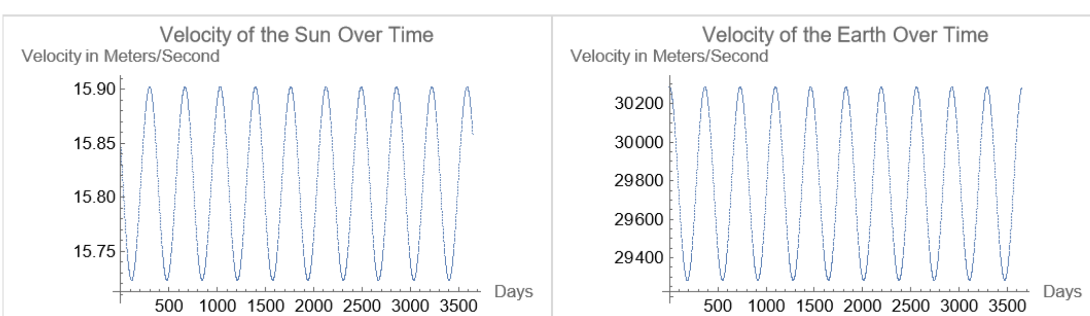
In[32]:=
```
simVel[body_, data_] := Drop[Table[Quantity[QuantityMagnitude[
     data[body, i * QuantityMagnitude[UnitConvert[ 1 day , "Seconds"]]]["Velocity"]],
    "Meters/Seconds"], {i, 0, Length[dates]}], -1];
sunSimVel2BodyICRS = simVel[sun, simulationData2BodyICRS];
earthSimVel2BodyICRS = simVel[earth, simulationData2BodyICRS];
```

In[35]:=
```
normData[data1_] := Table[Norm[data1⟦i⟧], {i, 1, Length[data1]}];
```

The plots below show the norms of the Sun's and Earth's velocities in the 2-body model over time. Similar to the distance graph, both of them show a sinusoidal pattern.

In[43]:=
```
GraphicsRow[MapThread[ListPlot[#1, AxesLabel → {"Days", "Velocity in Meters/Second"},
    PlotLabel → "Velocity of the " <> #2 <> " Over Time", ImageSize → Medium] &,
  {{normData[sunSimVel2BodyICRS], normData[earthSimVel2BodyICRS]}, {"Sun", "Earth"}}],
 Frame → All, FrameStyle → LightGray]
```

Out[•]=



Comparing the 2-Body Simulation With the Model Created Using AstroPosition

We will now compare the data from the simulation such as the position of the Sun and Earth and the distances between them with the real data from the AstroPosition model.

Getting the Data from the AstroPosition Model

To compare the two models, we first need to get the positions of the bodies from the AstroPosition model over the defined date range. We use the positionICRS function we defined earlier for this.

In[36]:=
```
sunPositionsICRS = Table[positionICRS[sun, i], {i, dates}];
earthPositionsICRS = Table[positionICRS[earth, i], {i, dates}];
```

We subtract these positions to get the distance between the Sun and the Earth in vector form, and we use the normDistances function we defined earlier to find the total distance between the Sun and the Earth over time.

In[38]:=
```
sunEarthDistancesICRS = earthPositionsICRS - sunPositionsICRS;
normDistancesEarthSunICRS = normDifferences[earthPositionsICRS, sunPositionsICRS];
```

After getting the position and distance data, we can get the velocities of the Sun and the Earth again with the same method used for initial velocities.

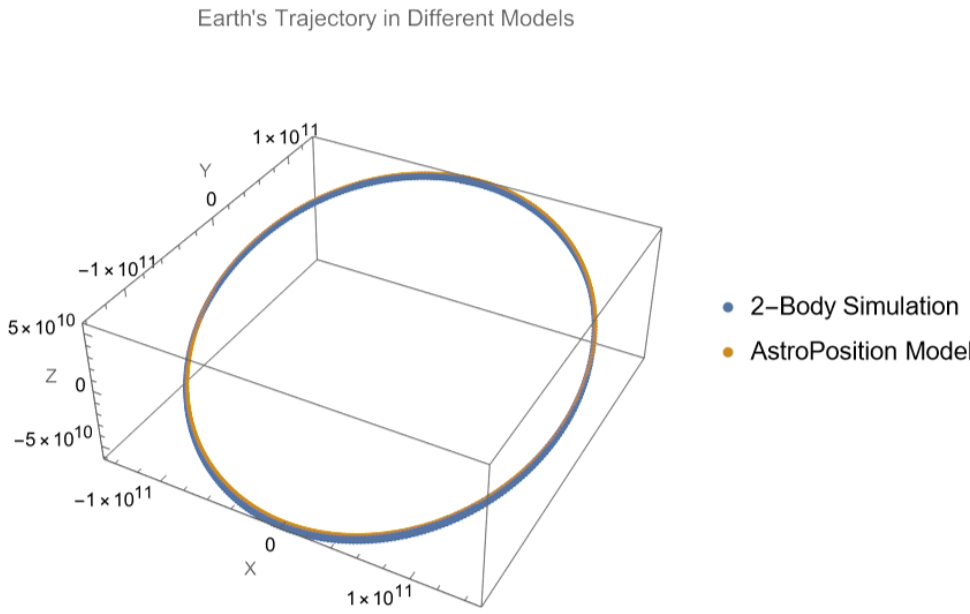
In[40]:=
```
sunVelocitiesICRS =
  (Table[positionICRS[sun, DatePlus[i, Quantity[0.5, "Minutes"]]], {i, dates}] -
    sunPositionsICRS) / UnitConvert[0.5 min , "Seconds"];
earthVelocitiesICRS =
  (Table[positionICRS[earth, DatePlus[i, Quantity[0.5, "Minutes"]]], {i, dates}] -
    earthPositionsICRS) / UnitConvert[0.5 min , "Seconds"];
```

Comparison of the 2-Body Model With the AstroPosition Model

We will now compare the data obtained from different simulations. We first plot the position of the Earth and Sun over 10 years, both from AstroPosition and simulation. In this plot, while the coordinate values seem to align well, we will iterate this graph with individual X, Y and Z-Coordinates for a detailed look.

In[46]:=
```
ListPointPlot3D[{earthSimPos2BodyICRS, earthPositionsICRS},
 PlotRange → All, PlotLegends → {"2-Body Simulation", "AstroPosition Model"},
 AxesLabel → {"X", "Y", "Z"}, PlotLabel → "Earth's Trajectory in Different Models"]
```

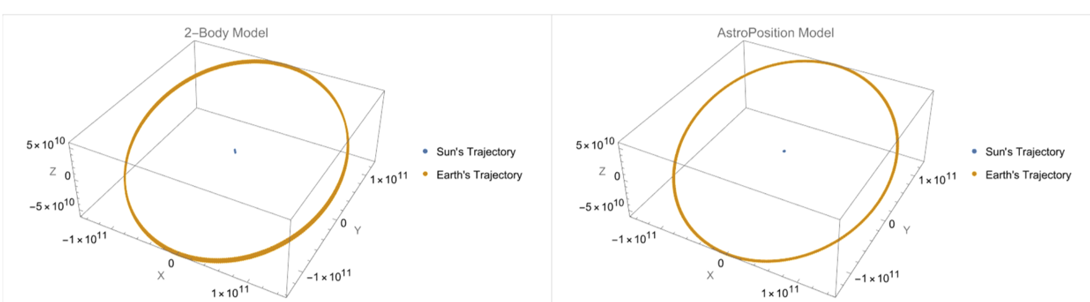
Out[•]=



Now we plot the 2-body systems separately as the 2-Body simulation and AstroPosition model. Both of them shows Earth's nearly circular orbit around the Sun. However, the differences between the data from the simulation can be seen due to the width of the line that represents the Earth's trajectories. As the line that represents Earth's trajectory in the 2-body model is thicker than the one in the AstroPosition, there is a deviation from the real data in the 2-body model.

In[88]:=
```
GraphicsRow[MapThread[ListPointPlot3D[#1, PlotRange → All,
    PlotLegends → {"Sun's Trajectory", "Earth's Trajectory"},
    AxesLabel → {"X", "Y", "Z"}, PlotLabel → #2, ImageSize → {350, 250}] &,
  {{{sunSimPos2BodyICRS, earthSimPos2BodyICRS}, {sunPositionsICRS, earthPositionsICRS}},
   {"2-Body Model", "AstroPosition Model"}}],
 Frame → All, FrameStyle → LightGray, ImageSize → {1000, 300}]
```

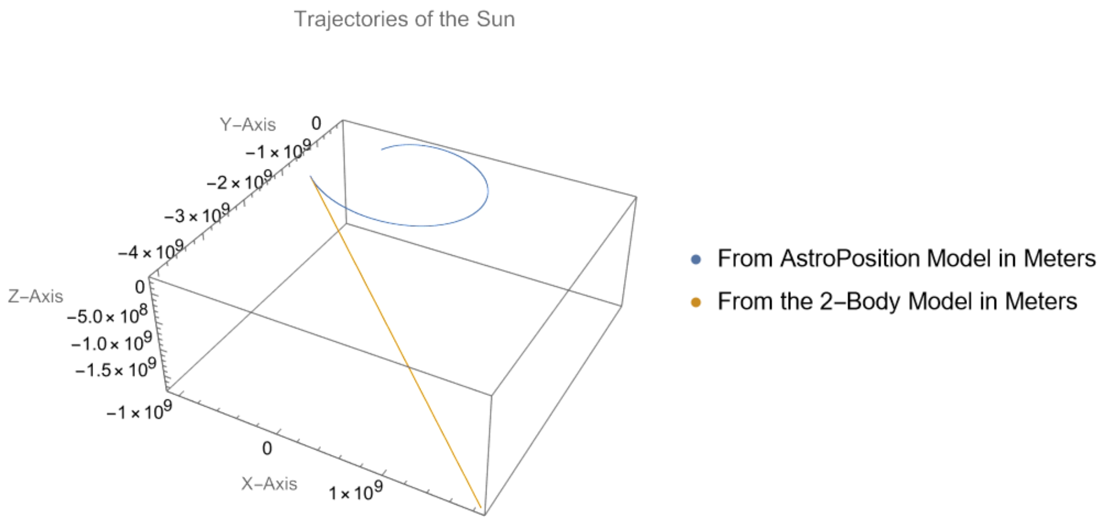
Out[•]=



As the position of the Sun wasn't visible in the plot above due to scaling, we plot it here again only for Sun's positions from AstroPosition and simulation. Even though both start from {0,0,0} and go further away from this point with time, Sun's trajectory seems to be linear in the simulation while in AstroPosition, it's a more elliptical shape.

```
In[71]:= ListPointPlot3D[{sunPositionsICRS, sunSimPos2BodyICRS},
  PlotLegends → {"From AstroPosition Model in Meters", "From the 2-Body Model in Meters"},
  AxesLabel → {"X-Axis", "Y-Axis", "Z-Axis"}, PlotLabel → "Trajectories of the Sun"]
```

Out[•]=

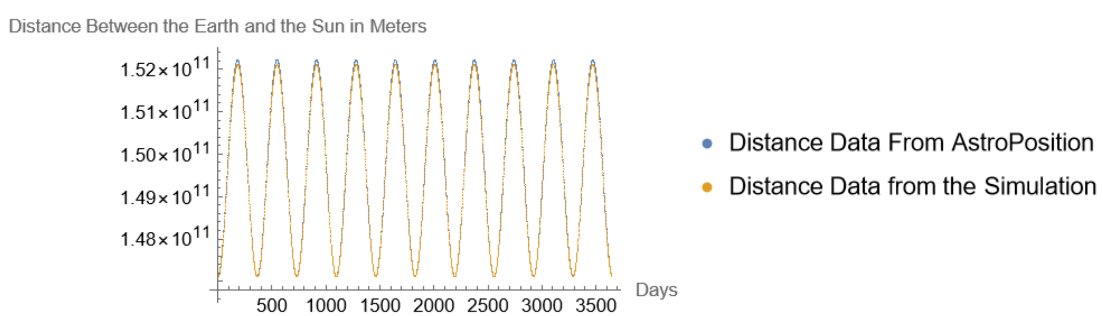


This discrepancy can be attributed to the lack of other objects exerting their gravitational force on the Sun. As we add more objects, the simulation becomes more accurate. For example Jupiter has a significant influence on the Sun and other planets and our research has shown that with the introduction of Jupiter to the Sun-Moon-Earth system, the simulation data and actual data are quite similar.

We will now plot the distance between the Earth and the Sun with the data from AstroPosition and our simulation. Even though they generally overlap, some differences can be seen in the maximum points. That might be partly due the fact that there would be more errors in value for the maximum points even if the error rate remained the same in percentage, as the values are larger here for the maximum points.

```
In[72]:= ListPlot[{normData[sunEarthSimDistances2BodyICRS], normDistancesEarthSunICRS},
  PlotLegends → {"Distance Data From AstroPosition", "Distance Data from the Simulation"},
  AxesLabel → {"Days", "Distance Between the Earth and the Sun in Meters"}]
```

Out[•]=



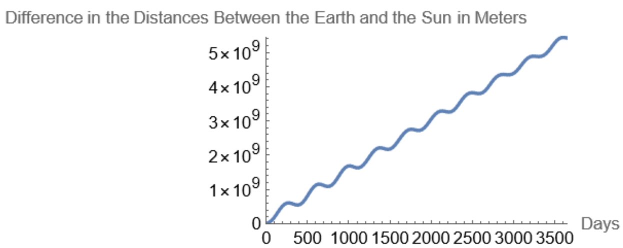
We get the differences of distance data from AstroPosition and simulation on the same date and plot the absolute value of the differences by defining another function. We get the absolute value as we are only interested in the magnitude of the differences.

In[52]:=
```
differences2BodyICRS =
  normDifferences[sunEarthDistancesICRS, sunEarthSimDistances2BodyICRS];
```

As the distance between the Earth and Sun are factors of 10^{11} from the graph above, getting factors of 10^{9} at most for the differences in distances in the graph below suggests a factor of 10^{-2} difference in the position data from AstroPosition and Simulation. However, this difference increases over time with fluctuations. Hence, after a while, this difference might reach a more significant ratio.

In[74]:=
```
ListLinePlot[differences2BodyICRS, AxesLabel →
  {"Days", "Difference in the Distances Between the Earth and the Sun in Meters"},
 PlotRange → {{0, Length[differences2BodyICRS]},
   Map[QuantityMagnitude, {Min[differences2BodyICRS], Max[differences2BodyICRS]}]}]
```

Out[•]=



This graph shows us that with time the differences in distances between the Earth and the Sun grows. The drift can be attributed to many factors, but we don't know for certain why we see this phenomenon. For example there could be model drift where the other planets, such as Jupiter, that we haven't accounted for.

Below, we calculate the specific average error rate for the distances between the Earth and the Sun comparing the 2-body and AstroPosition model. By getting the ratio of the differences to the data from AstroPosition and its average, the mean error rate is calculated to be around 1.85%.

In[75]:=
```
averageErrorDistances =
 Quantity[Mean[(differences2BodyICRS / normDistancesEarthSunICRS) * 100], "Percent"]
```

Out[•]=

1.85235%

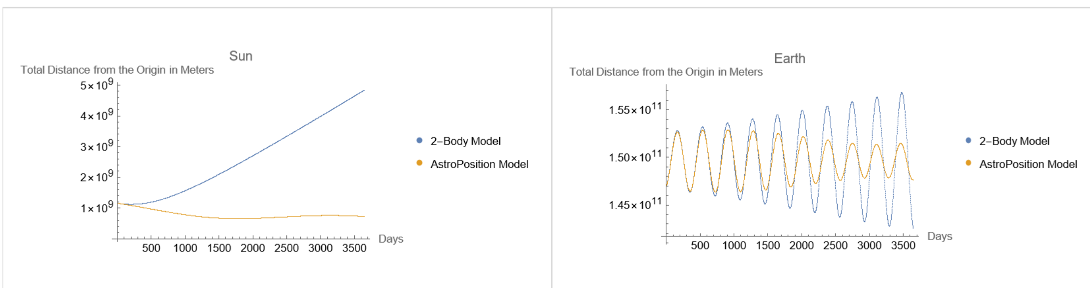
As we plot the position of the Sun, it can be seen that some of the differences in the distance between the Sun and the Earth come from the Sun's position points. Even though, the position of the Sun from AstroPosition shows an elliptical trajectory, the distance from the origin gets larger over time according to the data from the simulation which could explain the differences in the distances between the Earth and the Sun. Here, we use the previously defined function normData to get the net positions /distances from the origin of the bodies.

When we plot the net position of the Sun over time both for the AstroPosition and 2-body model, it can be seen that although the difference in the distance between the Sun and the Earth was around 1.85%, the total distance of the Sun gets multiplied over time reaching 5 times of the value from the AstroPosition at the end in the 2-body model.

Unlike the Sun's position, the total distance of the Earth from the origin seems to hold at the beginning, but the difference between the 2-body model and AstroPosition increases with time.

In[87]:=
```
GraphicsRow[MapThread[ListPlot[#1, PlotLegends → {"2-Body Model", "AstroPosition Model"},
    AxesLabel → {"Days", "Total Distance from the Origin in Meters"},
    PlotLabel → #2, ImageSize → {350, 250}] &,
  {{{normData[sunSimPos2BodyICRS], normData[sunPositionsICRS]},
    {normData[earthSimPos2BodyICRS], normData[earthPositionsICRS]}}, {"Sun", "Earth"}}],
 Frame → All, FrameStyle → LightGray, ImageSize → {1000, 300}]
```

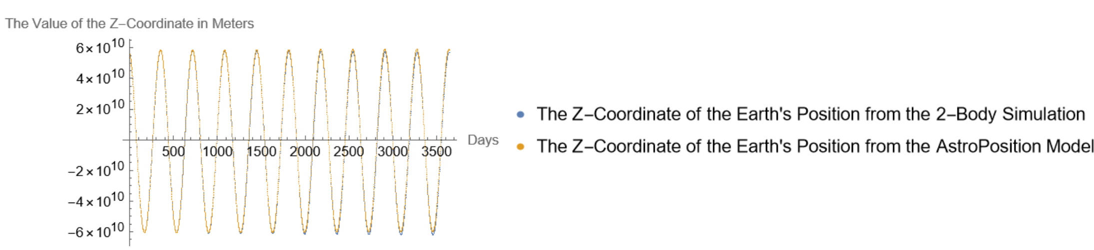
Out[∘]=



When we plot the Z-coordinates of Earth's positions from both models, such difference between the models isn't visible as the values are now smaller.

In[77]:=
```
ListPlot[{Table[earthSimPos2BodyICRS〚n, 3〛, {n, Length[earthSimPos2BodyICRS]}],
  Table[earthPositionsICRS〚n, 3〛, {n, Length[earthPositionsICRS]}]},
 PlotLegends → {"The Z-Coordinate of the Earth's Position from the 2-Body Simulation",
   "The Z-Coordinate of the Earth's Position from the AstroPosition Model"},
 AxesLabel → {"Days", "The Value of the Z-Coordinate in Meters"}]
```

Out[∘]=

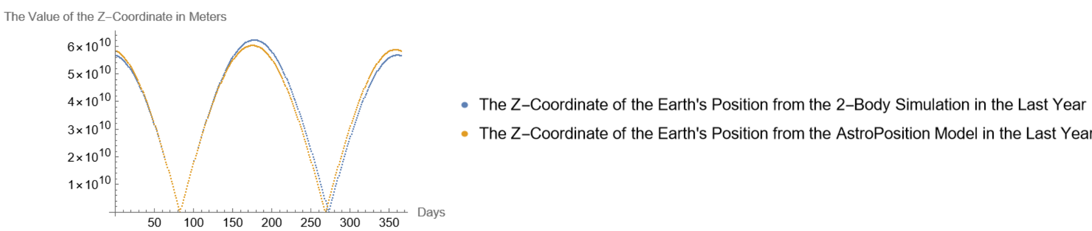


However, to get a closer look, we also plot the Z-coordinate of the position of the Earth only in the tenth/last year. For this we define a function called lastYear. When we plot these positions, a difference of 10^9 factor can be seen this time which seems to hold with the maximum error in the total distance plot.

In[73]:=
```
lastYear[data_] := Table[data[[i]], {i, Length[data] - 365, Length[data]}];
earthSimPos2BodyICRSLastYear = lastYear[earthSimPos2BodyICRS[[All, 3]]];
earthPositionsICRSLastYear = lastYear[earthPositionsICRS[[All, 3]]];
```

In[81]:=
```
ListPlot[{normData[earthSimPos2BodyICRSLastYear],
  normData[earthPositionsICRSLastYear]}, PlotLegends →
  {"The Z-Coordinate of the Earth's Position from the 2-Body Simulation in the
     Last Year", "The Z-Coordinate of the Earth's Position
     from the AstroPosition Model in the Last Year"},
 AxesLabel → {"Days", "The Value of the Z-Coordinate in Meters"}]
```

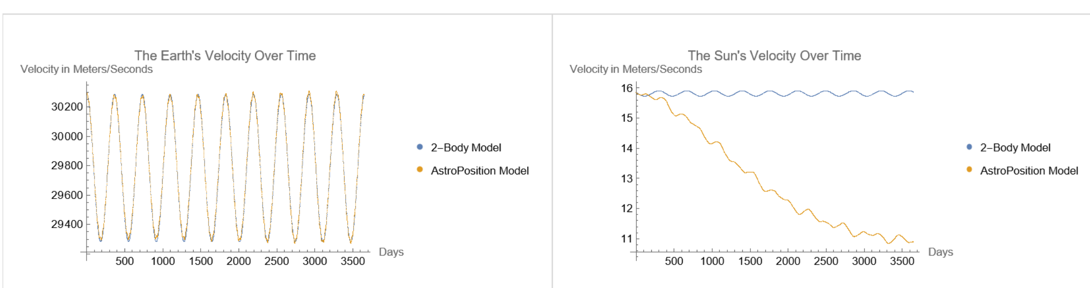
Out[•]=



This time we compare the velocities of each body in different simulations. Below, we plot the velocities of the Earth and the Sun separately. Even though the velocity of the Earth in the 2-body simulation seems to align with the AstroPosition model, the difference between the velocity of the Sun in different models seems to increase over time. The Sun experiences more perturbations due to other bodies that we didn’t include in the 2-body model. Hence, this can suggest that while the Sun is the greatest factor on the velocity and position of the Earth, the Sun is largely affected by other bodies. To investigate this effect, we will add the Moon to our simulation, creating the 3-body model.

In[86]:=
```
GraphicsRow[
 MapThread[ListPlot[#1, PlotLegends → {"2-Body Model", "AstroPosition Model"}, AxesLabel →
     {"Days", "Velocity in Meters/Seconds"}, PlotLabel → #2, ImageSize → {350, 250}] &,
  {{{normData[earthSimVel2BodyICRS], normData[earthVelocitiesICRS]},
    {normData[sunSimVel2BodyICRS], normData[sunVelocitiesICRS]}},
   {"The Earth's Velocity Over Time", "The Sun's Velocity Over Time"}}],
 Frame → All, FrameStyle → LightGray, ImageSize → {1000, 300}]
```

Out[•]=



Building the 3-Body Model

Now, we will build a three-body simulation which is adding the Moon to the two-body simulations above.

Setting Up the Moon’s Initial Conditions

As the initial parameters of the Sun and the Earth are already defined, we only need to define the initial conditions of the Moon here like we did previously above.

In[89]:=
```
moon = Moon PLANETARY MOON ;
moonMass = Moon PLANETARY MOON [ mass ];
```

In[91]:=
```
initialMoonPositionICRS1 = positionICRS[moon, Sat 1 Jan 2000 00:00:00 UTC];
initialMoonPositionICRS2 = positionICRS[moon, Sat 1 Jan 2000 00:00:00 UTC + 0.5 min];
initialMoonVelocityICRS = (initialMoonPositionICRS2 - initialMoonPositionICRS1) /
   UnitConvert[0.5 min , "Seconds"];
moonPositionsICRS = Table[positionICRS[moon, i], {i, dates}];
```

Building the 3-Body Model and Obtaining Its Data

After defining all of the initial conditions, we build the 3-body simulation, again over 10 years. We will also check the simulation time for this model.

In[95]:=
```
simulationData3BodyICRS = NBodySimulation["Newtonian", <|sun → <|"Mass" → sunMass,
      "Position" → initialSunPositionICRS1, "Velocity" → initialSunVelocityICRS|>,
    earth → <|"Mass" → earthMass,
      "Position" → initialEarthPositionICRS1, "Velocity" → initialEarthVelocityICRS|>,
    moon → <|"Mass" → moonMass, "Position" → initialMoonPositionICRS1 ,
      "Velocity" → initialMoonVelocityICRS|>|>, 10 yr , MaxSteps → Infinity];
```

In[96]:=
```
simulationTime3BodyICRS = simulationData3BodyICRS["SimulationTime"]
```

Out[#]=

3.1536×10^8

After building the 3-body simulation, we get the position and velocity data from the simulation for each body along with the bodies’ distances to each other. For this, we first get the position of each body over time as a list.

In[97]:=
```
sunSimPos3BodyICRS =
  Table[Quantity[simulationData3BodyICRS[sun, "Position", t], "Meters"],
   {t, 0, simulationTime3BodyICRS, QuantityMagnitude[UnitConvert[ 1 day , "Seconds"]]}];
earthSimPos3BodyICRS =
  Table[Quantity[simulationData3BodyICRS[earth, "Position", t], "Meters"],
   {t, 0, simulationTime3BodyICRS, QuantityMagnitude[UnitConvert[ 1 day , "Seconds"]]}];
moonSimPos3BodyICRS =
  Table[Quantity[simulationData3BodyICRS[moon, "Position", t], "Meters"],
   {t, 0, simulationTime3BodyICRS, QuantityMagnitude[UnitConvert[ 1 day , "Seconds"]]}];
```

We subtract these values from each other to get the distance between each body in vector form.

In[100]:=
```
sunEarthSimDistances3BodyICRS = earthSimPos3BodyICRS - sunSimPos3BodyICRS;
sunMoonSimDistances3BodyICRS = moonSimPos3BodyICRS - sunSimPos3BodyICRS;
earthMoonSimDistances3BodyICRS = earthSimPos3BodyICRS - moonSimPos3BodyICRS;
```

Apart from the vector form, to be able to compare different models with each other numerically, we also get the magnitude of these distance vectors.

In[103]:=
```
normSimDistancesEarthSun3BodyICRS =
  normDifferences[earthSimPos3BodyICRS, sunSimPos3BodyICRS];
normSimDistancesMoonSun3BodyICRS =
  normDifferences[moonSimPos3BodyICRS, sunSimPos3BodyICRS];
normSimDistancesMoonEarth3BodyICRS =
  normDifferences[earthSimPos3BodyICRS, moonSimPos3BodyICRS];
```

As we will also compare velocities, we also get the velocity of each body over time.

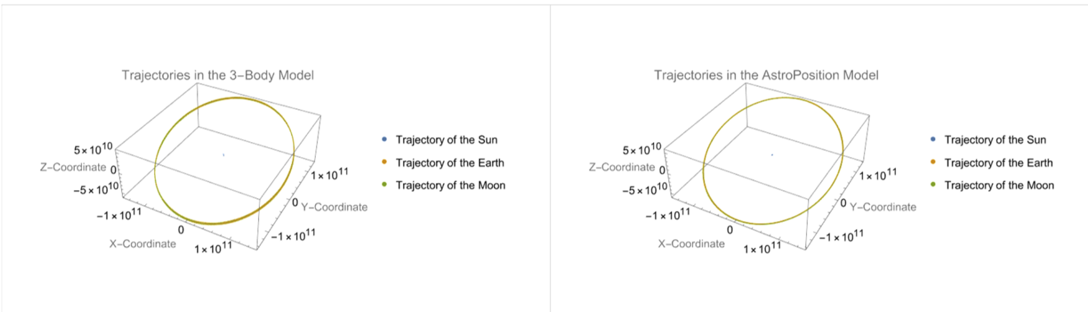
In[106]:=
```
sunSimVel3BodyICRS = simVel[sun, simulationData3BodyICRS];
earthSimVel3BodyICRS = simVel[earth, simulationData3BodyICRS];
moonSimVel3BodyICRS = simVel[moon, simulationData3BodyICRS];
```

We plot the trajectories of the Sun, Earth and Moon here. While the 3-body model shows elliptical orbits which seem to align with the data from AstroPosition for the Moon and the Earth, Sun’s trajectory seems to be more linear whereas Sun has an elliptical trajectory in the data from AstroPosition. This difference may have several including ignoring the effects of Jupiter due to its huge mass although it is further away from the Earth. Hence, in the next section, we will create a 4-body model to account for the changes Jupiter creates. Below, we plot the trajectories of each body but models separately. The general shapes of the plots seem to suggest they align well.

In[111]:=

```
GraphicsRow[MapThread[
  ListPointPlot3D[#1, PlotLegends → {"Trajectory of the Sun", "Trajectory of the Earth",
      "Trajectory of the Moon"}, PlotLabel → #2, AxesLabel →
     {"X-Coordinate", "Y-Coordinate", "Z-Coordinate"}, ImageSize → {300, 250} ] &,
  {{{sunSimPos3BodyICRS, earthSimPos3BodyICRS, moonSimPos3BodyICRS},
    {sunPositionsICRS, earthPositionsICRS, moonPositionsICRS}},
   {"Trajectories in the 3-Body Model", "Trajectories in the AstroPosition Model"}}],
 Frame → All, FrameStyle → LightGray, ImageSize → {1000, 300}]
```

Out[•]=

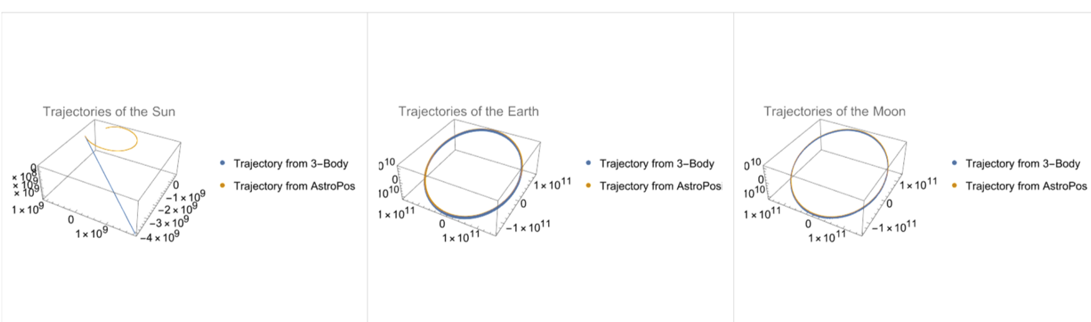


To get a closer look of the bodies, we plot them alone here both with the 3-body and AstroPosition model.

In[461]:=

```
GraphicsRow[MapThread[ListPointPlot3D[#1,
    PlotLegends → {"Trajectory from 3-Body", "Trajectory from AstroPosition"},
    PlotLabel → #2, ImageSize → {200, 300}] &,
  {{{sunSimPos3BodyICRS, sunPositionsICRS}, {earthSimPos3BodyICRS, earthPositionsICRS},
    {moonSimPos3BodyICRS, moonPositionsICRS}}, {"Trajectories of the Sun",
    "Trajectories of the Earth", "Trajectories of the Moon"}}],
 Frame → All, FrameStyle → LightGray, ImageSize → {1000, 320}]
```

Out[•]=



The most visible difference is in the trajectory of the Sun. The Sun seems to move linearly in the 3-body model like in the 2-body model even when we added the Moon. Clearly the Moon doesn't seem to have much effect due to its low mass. Also note that the the Moon and the Earth have similar trajectories. This is because the sun is perturbing the Moon too. To investigate the reason for why this difference occurs, we will also add the Jupiter into our system.

Investigating Jupiter’s Effect on the 3-Body Simulation - Building the 4-Body Model

We will now be adding Jupiter to estimate its effect on the 3-body simulation, creating the 4-body and last model.

Setting Up Jupiter’s Initial Conditions

As the other initial parameters are already defined, we only define Jupiter’s initial parameters here. However, we won’t get Jupiter’s data over 10 years from AstroPosition as we won’t use Jupiter in the comparisons.

In[122]:=

```
jupiter = Jupiter PLANET ;
jupiterMass = Jupiter PLANET [ mass ];
initialJupiterPositionICRS1 = positionICRS[jupiter, Sat 1 Jan 2000 00:00:00 UTC ];
initialJupiterPositionICRS2 = positionICRS[jupiter, Sat 1 Jan 2000 00:00:00 UTC + 0.5 min ];
initialJupiterVelocityICRS =
  (initialJupiterPositionICRS2 - initialJupiterPositionICRS1) /
   UnitConvert[ 0.5 min , "Seconds"];
```

Building the 4-Body Model and Getting the Data

After these definitions, we can build the 4-body simulation.

In[127]:=

```
simulationData4BodyICRS  = NBodySimulation["Newtonian", <|sun → <|"Mass" → sunMass,
      "Position" → initialSunPositionICRS1, "Velocity" → initialSunVelocityICRS|>,
    earth → <|"Mass" → earthMass, "Position" → initialEarthPositionICRS1,
      "Velocity" → initialEarthVelocityICRS|>, moon → <|"Mass" → moonMass,
      "Position" → initialMoonPositionICRS1 , "Velocity" → initialMoonVelocityICRS|>,
    jupiter → <|"Mass" → jupiterMass, "Position" → initialJupiterPositionICRS1,
      "Velocity" → initialJupiterVelocityICRS|>|>, 10 yr , MaxSteps → Infinity];
```

In[128]:=

```
simulationTime4BodyICRS = simulationData4BodyICRS["SimulationTime"]
```

Out[#]=

3.1536×10^{8}

We get the positions and velocities of each body just like how we did for the 2 and 3-body simulation.

We can also get the data for Jupiter, however, as our focus is the dynamics of the Earth-Sun-Moon system, we won't need Jupiter's data.

In[129]:=

```
sunSimPos4BodyICRS =
  Table[Quantity[simulationData4BodyICRS[sun, "Position", t], "Meters"],
   {t, 0, simulationTime4BodyICRS, QuantityMagnitude[UnitConvert[ 1 day , "Seconds"]]}];
earthSimPos4BodyICRS =
  Table[Quantity[simulationData4BodyICRS[earth, "Position", t], "Meters"],
   {t, 0, simulationTime4BodyICRS, QuantityMagnitude[UnitConvert[ 1 day , "Seconds"]]}];
moonSimPos4BodyICRS =
  Table[Quantity[simulationData4BodyICRS[moon, "Position", t], "Meters"],
   {t, 0, simulationTime4BodyICRS, QuantityMagnitude[UnitConvert[ 1 day , "Seconds"]]}];
```

Here, we get the distances between each bodies in the 4-body simulation over time in vector form.

In[132]:=

```
sunEarthSimDistances4BodyICRS = earthSimPos4BodyICRS - sunSimPos4BodyICRS;
sunMoonSimDistances4BodyICRS = moonSimPos4BodyICRS - sunSimPos4BodyICRS;
earthMoonSimDistances4BodyICRS = earthSimPos4BodyICRS - moonSimPos4BodyICRS;
```

We also define the magnitude of the distance between them.

In[135]:=

```
normSimDistancesEarthSun4BodyICRS =
  normDifferences[earthSimPos4BodyICRS, sunSimPos4BodyICRS];
normSimDistancesMoonSun4BodyICRS =
  normDifferences[moonSimPos4BodyICRS, sunSimPos4BodyICRS];
normSimDistancesMoonEarth4BodyICRS =
  normDifferences[earthSimPos4BodyICRS, moonSimPos4BodyICRS];
```

Below, we get the velocity of each body using the simVel function.

In[138]:=

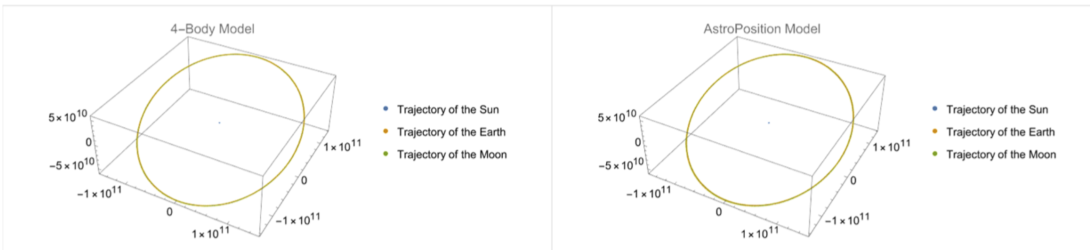
```
sunSimVel4BodyICRS = simVel[sun, simulationData4BodyICRS];
earthSimVel4BodyICRS = simVel[earth, simulationData4BodyICRS];
moonSimVel4BodyICRS = simVel[moon, simulationData4BodyICRS];
```

We now again plot the models in 3D and the 4-body model also shows elliptical trajectories for the Earth and the Moon.

In[151]:=

```
GraphicsRow[
 MapThread[ListPointPlot3D[#1, PlotLabel → #2, ImageSize → {300, 200}, PlotLegends →
      {"Trajectory of the Sun", "Trajectory of the Earth", "Trajectory of the Moon"}] &,
   {{{sunSimPos4BodyICRS, earthSimPos4BodyICRS, moonSimPos4BodyICRS},
     {sunPositionsICRS, earthPositionsICRS, moonPositionsICRS}},
    {"4-Body Model", "AstroPosition Model"}}], Frame → All,
 FrameStyle → LightGray, ImageSize → {1000, 300}]
```

Out[∘]=

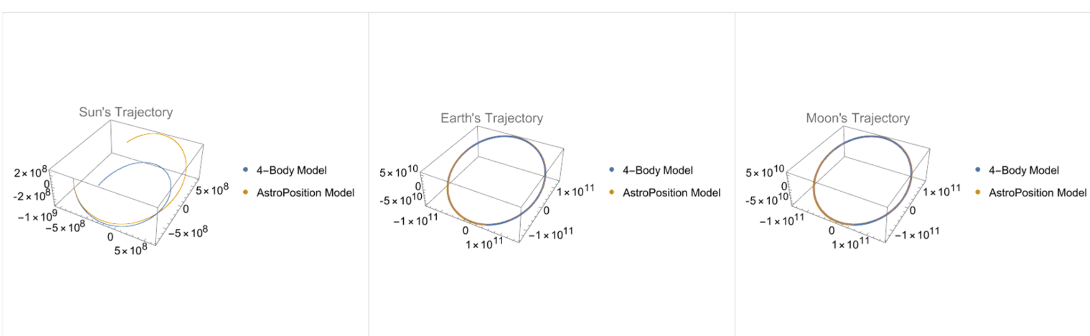


Once again we will plot the side by side comparisons of the 4 body model and the AstroPosition Model

In[462]:=

```
GraphicsRow[
 MapThread[ListPointPlot3D[#1, PlotLegends → {"4-Body Model", "AstroPosition Model"},
     PlotLabel → #2, ImageSize → {200, 300}] &,
   {{{sunSimPos4BodyICRS, sunPositionsICRS}, {earthSimPos4BodyICRS, earthPositionsICRS},
     {moonSimPos4BodyICRS, moonPositionsICRS}},
    {"Sun's Trajectory", "Earth's Trajectory", "Moon's Trajectory"}}],
 Frame → All, FrameStyle → LightGray, ImageSize → {1000, 310}]
```

Out[∘]=



When we plot the bodies individually, Sun has the biggest visible difference in its trajectory compared to the 3-body model. While the Sun had nearly a linear trajectory in the 3-body model, it shows a curved trajectory here which suggests the values of its positions are closer to AstroPosition now. Hence, we can suggest even though Jupiter’s far away, its big mass seems to improve the accuracy of the model. Moreover we can see that if we add the other gas giants our model will become more and more accurate.

Comparison of Different Models

We have built all of the models we will use in the previous sections. We will now compare their data to each other with plots and datasets and see what conclusions we can draw.

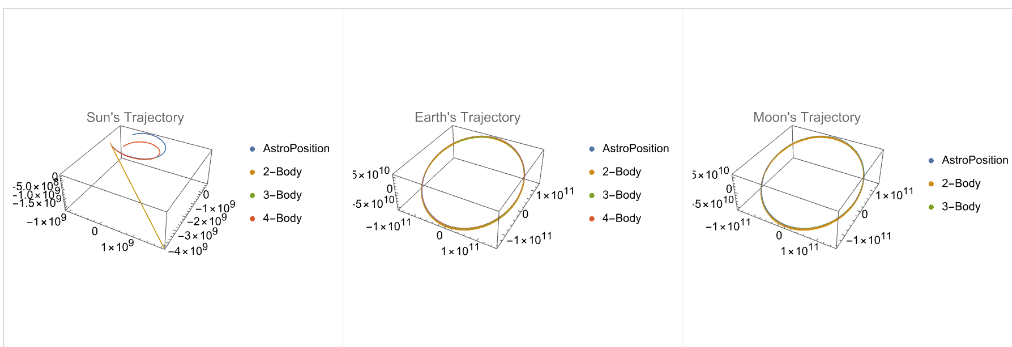
Positions in the ICRS Model

We will first compare the positions of the Sun, Earth and Moon in different models. The plots below compare different models for each body separately.

In[153]:=

```
GraphicsRow[MapThread[ListPointPlot3D[#1,
    PlotLegends → Which[Length[#1] == 4, {"AstroPosition", "2-Body", "3-Body", "4-Body"},
      Length[#1] == 3, {"AstroPosition", "2-Body", "3-Body"}, True, ""],
    PlotLabel → #2, ImageSize → {200, 305}] &,
   {{{sunPositionsICRS, sunSimPos2BodyICRS, sunSimPos3BodyICRS, sunSimPos4BodyICRS},
     {earthPositionsICRS, earthSimPos2BodyICRS,
      earthSimPos3BodyICRS, earthSimPos4BodyICRS},
     {moonPositionsICRS, moonSimPos3BodyICRS, moonSimPos4BodyICRS}},
    {"Sun's Trajectory", "Earth's Trajectory", "Moon's Trajectory"}}],
  Frame → All, FrameStyle → LightGray, ImageSize → {1000, 310}]
```

Out[•]=



Note that the data from some of the model seem to overlap like Sun’s trajectory for the 2 and 3-body models. We clearly see a visual that as we increase the amount of bodies in the system, the trajectory of the sun will get closer and closer to the correct value. For the Earth and the Moon’s trajectories, all of the models seem to overlap with elliptical orbits meaning that our simulation was very accurate.

For a closer look, we investigate the 2D plots of the bodies’ positions with net distance from the origin and its components.

In[206]:=

```
GraphicsGrid[
 Partition[MapThread[ListPlot[#1, PlotLegends → {"AstroPosition Model", "2-Body Model",
       "3-Body Model", "4-Body Model"}, PlotLabel → #2, ImageSize → {300, 350}] &,
    {{{normData[earthPositionsICRS], normData[earthSimPos2BodyICRS],
       normData[earthSimPos3BodyICRS], normData[earthSimPos4BodyICRS]},
      {earthPositionsICRS⟦All, 3⟧, earthSimPos2BodyICRS⟦All, 3⟧,
       earthSimPos3BodyICRS⟦All, 3⟧, earthSimPos4BodyICRS⟦All, 3⟧},
      {earthPositionsICRS⟦All, 2⟧, earthSimPos2BodyICRS⟦All, 2⟧,
       earthSimPos3BodyICRS⟦All, 2⟧, earthSimPos4BodyICRS⟦All, 2⟧},
      {earthPositionsICRS⟦All, 1⟧, earthSimPos2BodyICRS⟦All, 1⟧,
       earthSimPos3BodyICRS⟦All, 1⟧, earthSimPos4BodyICRS⟦All, 1⟧}},
     {"Total Magnitude of the Earth's Position Over Time",
      "The Value of the Z-Coordinate of the Earth's Position Over Time",
      "The Value of the Y-Coordinate of the Earth's Position Over Time",
      "The Value of the X-Coordinate of the Earth's Position Over Time"}}], 2],
 Frame → All, FrameStyle → LightGray, ImageSize → {950, 780}]
```

Out[=]=

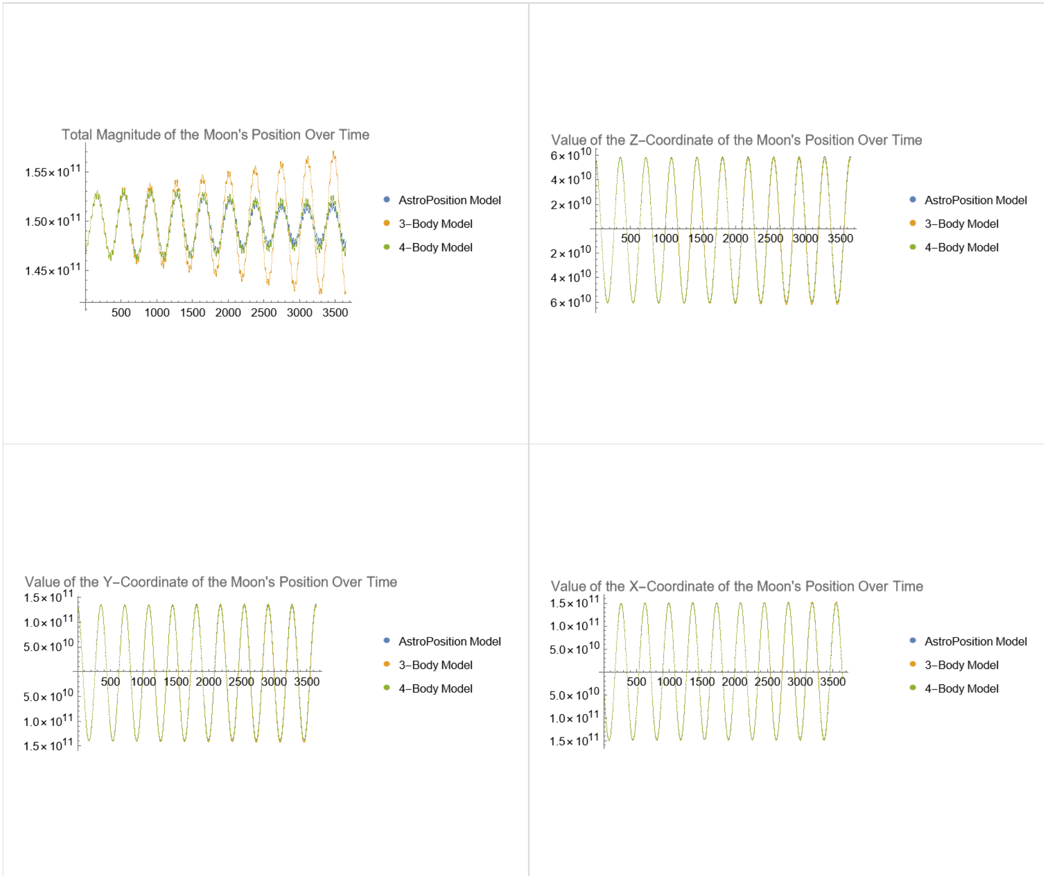
From these plots, we see that the biggest deviation from the simulations and the data in the AstroPosition was in the 2-body model, which would make sense as we ignored the bodies in it. Moreover we can observe that the 3-body model graph and the 2-body model graph are so similar that the 3-body model graph hides the 2-body model graph. This suggests the effect of the Moon is rather trivial. It is also interesting that the number of bodies may not directly correlate with less error. For example there were no significant improvements from the 2-body graph to the 3-body graph. From this we might be able to conclude that the mass and gravitational effect of the object matters more than the number of bodies in the simulation.

Similarly, even though the magnitude of the Moon's position over time has a sinusoidal pattern in all of the models, the difference between the 3-body and AstroPosition model gets larger over time.

In[180]:=

```
GraphicsGrid[Partition[MapThread[
   ListPlot[#1, PlotLegends → {"AstroPosition Model", "3-Body Model", "4-Body Model"},
     PlotLabel → #2, ImageSize → {300, 350}] &,
   {{{normData[moonPositionsICRS], normData[moonSimPos3BodyICRS],
       normData[moonSimPos4BodyICRS]}, {moonPositionsICRS⟦All, 3⟧,
       moonSimPos3BodyICRS⟦All, 3⟧, moonSimPos4BodyICRS⟦All, 3⟧},
     {moonPositionsICRS⟦All, 2⟧, moonSimPos3BodyICRS⟦All, 2⟧,
       moonSimPos4BodyICRS⟦All, 2⟧}, {moonPositionsICRS⟦All, 1⟧,
       moonSimPos3BodyICRS⟦All, 1⟧, moonSimPos4BodyICRS⟦All, 1⟧}},
    {"Total Magnitude of the Moon's Position Over Time",
     "The Value of the Z-Coordinate of the Moon's Position Over Time",
     "The Value of the Y-Coordinate of the Moon's Position Over Time",
     "The Value of the X-Coordinate of the Moon's Position Over Time"}}], 2],
 Frame → All, FrameStyle → LightGray, ImageSize → {950, 800}]
```

Out[*]=



It is quite interesting to see that when we take the magnitude of the X, Y, Z coordinates of the Moon's position over time, the four body model is quite similar to the AstroPosition data. As we've explained previously before, this is likely due to the fact that Jupiter was added making the simulation more realistic. On the other hand it is also interesting that AstroPosition model, the 3-body model and the 4-body model perform around the same for the individual graphs for the X, Y, Z coordinates of the Moon. This could be because the error for each of the X, Y, Z component are small but when we take the magnitude the grow large.

Instead of estimating the differences between the models from the position plots, we create numerical lists for the differences in the positions between each model. This way we can understand what exactly the graphs mean numerically. Below, we define the differences between 2-body, 3-body, 4-body from the AstroPosition data of the Sun, Moon and the Earth.

In[181]:=

```
differencesPosSun4BodyAstroICRS = normDifferences[sunSimPos4BodyICRS, sunPositionsICRS];
differencesPosSun3BodyAstroICRS = normDifferences[sunSimPos3BodyICRS, sunPositionsICRS];
differencesPosSun2BodyAstroICRS = normDifferences[sunSimPos2BodyICRS, sunPositionsICRS];
differencesPosEarth4BodyAstroICRS =
  normDifferences[earthSimPos4BodyICRS, earthPositionsICRS];
differencesPosEarth3BodyAstroICRS =
  normDifferences[earthSimPos3BodyICRS, earthPositionsICRS];
differencesPosEarth2BodyAstroICRS =
  normDifferences[earthSimPos2BodyICRS, earthPositionsICRS];
differencesPosMoon4BodyAstroICRS =
  normDifferences[moonSimPos4BodyICRS, moonPositionsICRS];
differencesPosMoon3BodyAstroICRS =
  normDifferences[moonSimPos3BodyICRS, moonPositionsICRS];
```

To be able to compare the systems numerically, we get the mean of the elements in the lists to get a single value using the Mean function. Since we will use the Mean multiple times, we define it as a function, modelsMeanDifferences.

In[189]:=

```
modelsMeanDifferences[differences_] := Map[Mean, differences];
modelsMeanPosDifferences4Body = modelsMeanDifferences[{differencesPosSun4BodyAstroICRS,
    differencesPosEarth4BodyAstroICRS, differencesPosMoon4BodyAstroICRS}]
modelsMeanPosDifferences3Body = modelsMeanDifferences[{differencesPosSun3BodyAstroICRS,
    differencesPosEarth3BodyAstroICRS, differencesPosMoon3BodyAstroICRS}]
modelsMeanPosDifferences2Body = modelsMeanDifferences[
  {differencesPosSun2BodyAstroICRS, differencesPosEarth2BodyAstroICRS}]
```

Out[ ]=

$\{2.91499 \times 10^{8}\ \text{m},\ 3.15636 \times 10^{8}\ \text{m},\ 3.15951 \times 10^{8}\ \text{m}\}$

Out[ ]=

$\{2.51275 \times 10^{9}\ \text{m},\ 2.51384 \times 10^{9}\ \text{m},\ 2.51385 \times 10^{9}\ \text{m}\}$

Out[ ]=

$\{2.51274 \times 10^{9}\ \text{m},\ 3.52761 \times 10^{9}\ \text{m}\}$

Although we found the mean absolute uncertainty/error above, to be able to compare these errors with the values from AstroPosition, we get the mean of the ratio of the corresponding elements in both lists.

In[193]:=

```
ratioDifferences[differences_, astropos_] :=
  MapThread[Quantity[Mean[(#1 / #2) * 100], "Percent"] &, {differences, astropos}];
ratiosPositions4Body =
 ratioDifferences[{differencesPosSun4BodyAstroICRS, differencesPosEarth4BodyAstroICRS,
   differencesPosMoon4BodyAstroICRS}, {normData[sunPositionsICRS],
   normData[earthPositionsICRS], normData[moonPositionsICRS]}]
ratiosPositions3Body =
 ratioDifferences[{differencesPosSun3BodyAstroICRS, differencesPosEarth3BodyAstroICRS,
   differencesPosMoon3BodyAstroICRS}, {normData[sunPositionsICRS],
   normData[earthPositionsICRS], normData[moonPositionsICRS]}]
ratiosPositions2Body =
 ratioDifferences[{differencesPosSun2BodyAstroICRS, differencesPosEarth2BodyAstroICRS},
  {normData[sunPositionsICRS], normData[earthPositionsICRS]}]
```

Out[ ]=

```
{40.7465%, 0.210685%, 0.210903%}
```

Out[ ]=

```
{352.62%, 1.68051%, 1.68052%}
```

Out[ ]=

```
{352.619%, 2.35555%}
```

We can visualize the data we got here with a data table to compare the accuracy of the models. The percentage uncertainties of the positions of the Earth and the Moon are always smaller than 2.5% with small (about 1%) changes with each model. However, the percentage uncertainties for the Sun are more significant with about 353% difference in the 2 and 3-body models, more than 3 times of the mean value from AstroPosition. But when we add Jupiter, creating the 4-body simulation, the error rate decreases to about 41%, which is more than 1/3 of the AstroPosition value. While this may seem like a significant amount of error, it is still much less than the previous error rate, showing the effect of the Jupiter on the system. This data table also explains the overlap of the 3D and 2D plots by showing the small differences for the Earth and the Moon as well as with 2 and 3-body models for the Sun’s positions.

In the data table below we will show the mean difference of the distance between each model to the AstroPosition Data and the error.

In[197]:=

```
dsPositions = Dataset[{Association["4-Body" → AssociationThread[
      {"Position of the Sun", "Position of the Earth", "Position of the Moon"} →
        Partition[Riffle[modelsMeanPosDifferences4Body, ratiosPositions4Body], 2]]],
   Association["3-Body" → AssociationThread[{"Position of the Sun",
        "Position of the Earth", "Position of the Moon"} → Partition[
         Riffle[modelsMeanPosDifferences3Body, ratiosPositions3Body], 2]]], Association[
    "2-Body" → AssociationThread[{"Position of the Sun", "Position of the Earth"} →
        Partition[Riffle[modelsMeanPosDifferences2Body, ratiosPositions2Body], 2]]]}]
```

Out[]=

4–Body	Position of the Sun	$\{2.91499\times10^{8}$ m, 40.7465%$\}$
	Position of the Earth	$\{3.15636\times10^{8}$ m, 0.210685%$\}$
	Position of the Moon	$\{3.15951\times10^{8}$ m, 0.210903%$\}$
3–Body	Position of the Sun	$\{2.51275\times10^{9}$ m, 352.62%$\}$
	Position of the Earth	$\{2.51384\times10^{9}$ m, 1.68051%$\}$
	Position of the Moon	$\{2.51385\times10^{9}$ m, 1.68052%$\}$
2–Body	Position of the Sun	$\{2.51274\times10^{9}$ m, 352.619%$\}$
	Position of the Earth	$\{3.52761\times10^{9}$ m, 2.35555%$\}$

Distances in the ICRS Model

After comparing the positions of each body, we will now compare the distances of the bodies to each other in different models. For this, we first need to define the differences between the bodies that we haven’t defined before.

In[199]:=

```
sunMoonDistancesICRS = moonPositionsICRS - sunPositionsICRS;
earthMoonDistancesICRS = earthPositionsICRS - moonPositionsICRS;
```

Then, we get the magnitudes of the distances instead of in the vector form.

In[201]:=

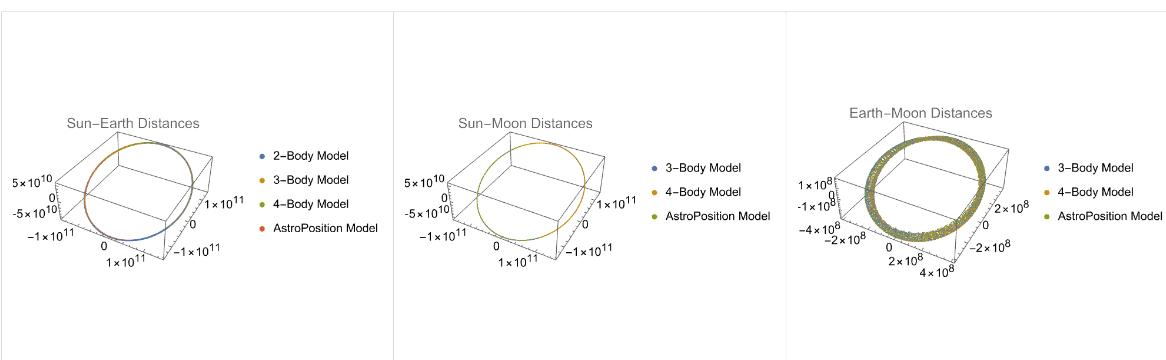
```
normDistancesMoonSunICRS = normDifferences[moonPositionsICRS, sunPositionsICRS];
normDistancesMoonEarthICRS = normDifferences[moonPositionsICRS, earthPositionsICRS];
```

Now we will draw the graphs that show the difference between the distances of the Earth, Moon, Sun and the four different models.

In[211]:=

```
GraphicsRow[MapThread[
  ListPointPlot3D[#1, PlotLegends → #2, PlotLabel → #3, ImageSize → {200, 300}] &,
  {{{sunEarthSimDistances2BodyICRS, sunEarthSimDistances3BodyICRS,
      sunEarthSimDistances4BodyICRS, sunEarthDistancesICRS},
    {sunMoonSimDistances3BodyICRS, sunMoonSimDistances4BodyICRS, sunMoonDistancesICRS},
    {earthMoonSimDistances3BodyICRS,
      earthMoonSimDistances4BodyICRS, earthMoonDistancesICRS}},
   {{"2-Body Model", "3-Body Model", "4-Body Model", "AstroPosition Model"},
    {"3-Body Model", "4-Body Model", "AstroPosition Model"},
    {"3-Body Model", "4-Body Model", "AstroPosition Model"}},
   {"Sun-Earth Distances", "Sun-Moon Distances", "Earth-Moon Distances"}}],
 Frame → All, FrameStyle → LightGray, ImageSize → {1000, 310}]
```

Out[=]=



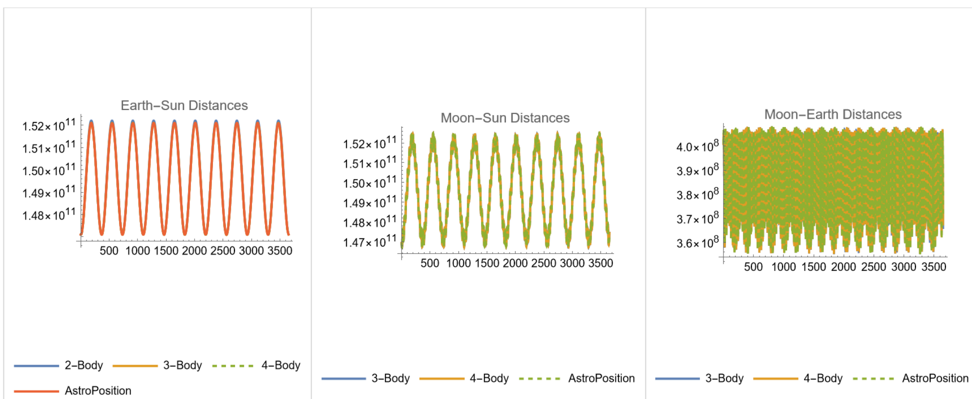
Even though the data in each plot seem to overlap, it can be seen there are more variations for the Earth-Moon distances as the plot looks more scattered. This is because the Moon’s motion is very complex and hard to predict due to the Sun’s perturbation force.

Now, we plot the norms of the distance data for the Earth-Sun, Earth-Moon and Moon-Sun distances.

In[226]:=

```
GraphicsRow[MapThread[ListLinePlot[#1, PlotLabel → #2,
    PlotLegends → Which[Length[#1] == 4, {"2-Body", "3-Body", "4-Body", "AstroPosition"},
      Length[#1] == 3, {"3-Body", "4-Body", "AstroPosition"}, True, ""],
    PlotStyle → {Default, Default, Dashed}, ImageSize → {240, 300}] &,
  {{{normSimDistancesEarthSun2BodyICRS, normSimDistancesEarthSun3BodyICRS,
     normSimDistancesEarthSun4BodyICRS, normDistancesEarthSunICRS},
    {normSimDistancesMoonSun3BodyICRS, normSimDistancesMoonSun4BodyICRS,
     normDistancesMoonSunICRS}, {normSimDistancesMoonEarth3BodyICRS,
     normSimDistancesMoonEarth4BodyICRS, normDistancesMoonEarthICRS}},
   {"Earth-Sun Distances", "Moon-Sun Distances", "Moon-Earth Distances"}}],
 Frame → All, FrameStyle → LightGray, ImageSize → {1025, 350}]
```

Out[=]=



All of them again show a sinusoidal pattern. It is interesting to note that the period of the Moon-Earth distances graph seems to be shorter than the other graphs, hence, making the graph look more packed. Hence, to get a closer look we define a function called last2Month.

In[352]:=

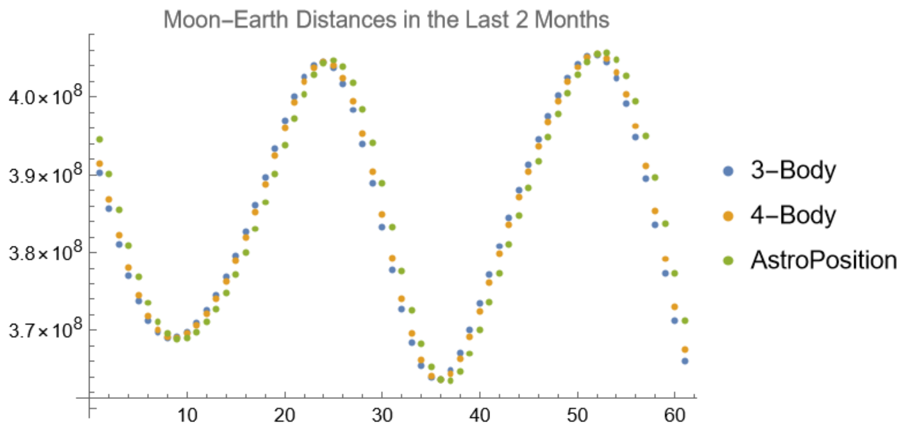
```
last2Month[data_] := Table[data⟦i⟧, {i, Length[data] - 60, Length[data]}];
```

When we use this function to get the distance between the Earth and the Moon for approximately the last 2 months, we can see that both 4-body and 3-body's patterns hold with the sinusoidal pattern of the AstroPosition model.

In[355]:=

```
ListPlot[{last2Month[normSimDistancesMoonEarth3BodyICRS],
  last2Month[normSimDistancesMoonEarth4BodyICRS],
  last2Month[normDistancesMoonEarthICRS]},
 PlotLegends → {"3-Body", "4-Body", "AstroPosition"},
 PlotLabel → "Moon-Earth Distances in the Last 2 Months"]
```

Out[∘]=



We can get the variances of the distance data from the AstroPosition model by getting the norm of the square of the differences in the distances of the model and AstroPosition.

In[244]:=

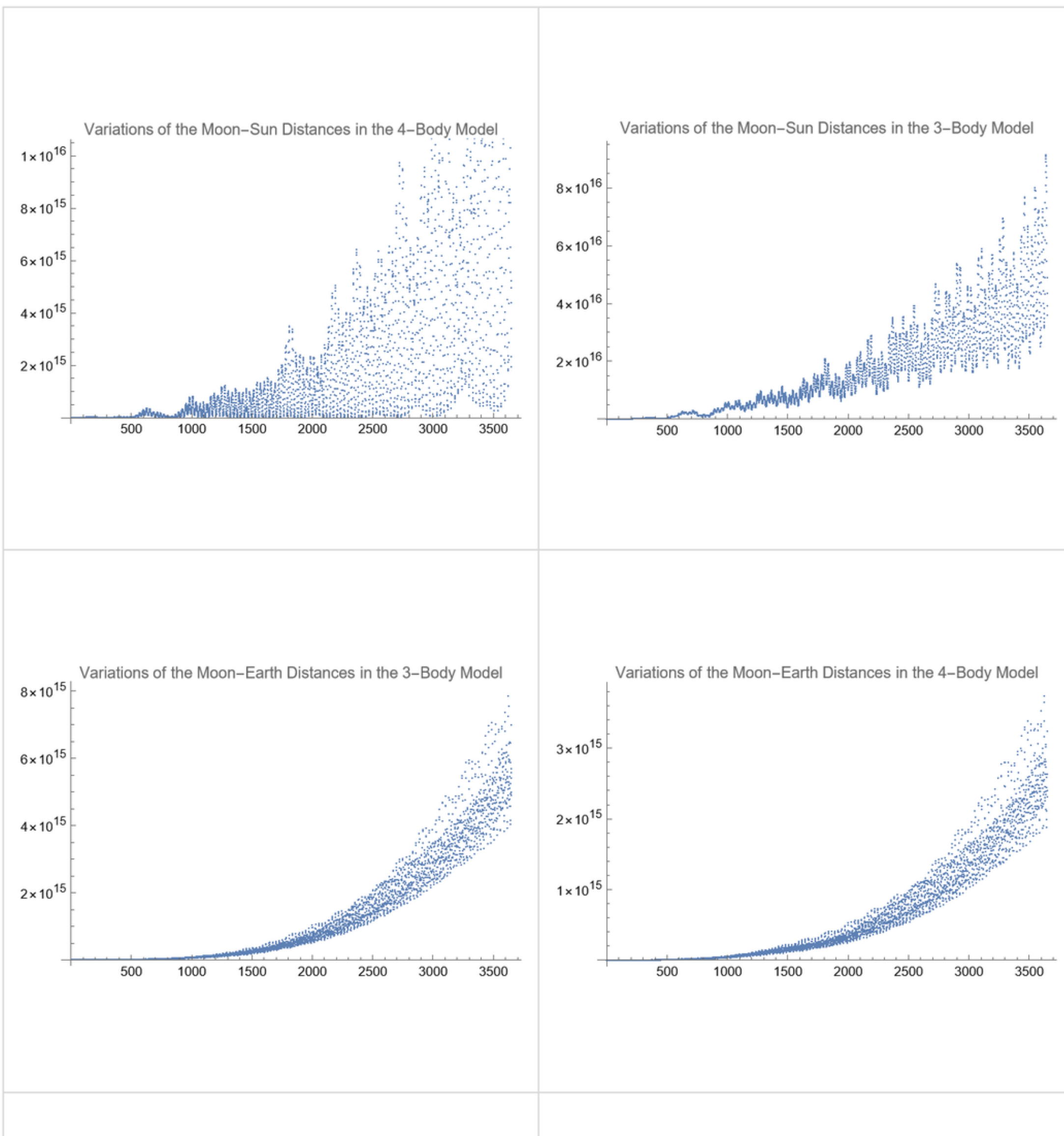
```
varMoonSun3BodyICRS =
  Table[Norm[(sunMoonSimDistances3BodyICRS[[i]] - sunMoonDistancesICRS[[i]])^2],
   {i, 1, Length[sunMoonDistancesICRS]}];
varMoonSun4BodyICRS =
  Table[Norm[(sunMoonSimDistances4BodyICRS[[i]] - sunMoonDistancesICRS[[i]])^2],
   {i, 1, Length[sunMoonDistancesICRS]}];
varMoonEarth3BodyICRS =
  Table[Norm[(earthMoonSimDistances3BodyICRS[[i]] - earthMoonDistancesICRS[[i]])^2],
   {i, 1, Length[earthMoonDistancesICRS]}];
varMoonEarth4BodyICRS =
  Table[Norm[(earthMoonSimDistances4BodyICRS[[i]] - earthMoonDistancesICRS[[i]])^2],
   {i, 1, Length[earthMoonDistancesICRS]}];
varEarthSun2BodyICRS =
  Table[Norm[(sunEarthSimDistances2BodyICRS[[i]] - sunEarthDistancesICRS[[i]])^2],
   {i, 1, Length[sunEarthDistancesICRS]}];
varEarthSun3BodyICRS =
  Table[Norm[(sunEarthSimDistances3BodyICRS[[i]] - sunEarthDistancesICRS[[i]])^2],
   {i, 1, Length[sunEarthDistancesICRS]}];
varEarthSun4BodyICRS =
  Table[Norm[(sunEarthSimDistances4BodyICRS[[i]] - sunEarthDistancesICRS[[i]])^2],
   {i, 1, Length[sunEarthDistancesICRS]}];
```

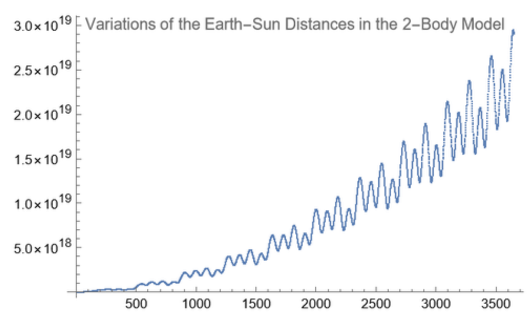
Now we will create plots to see the variations between the models and the AstroPosition data.

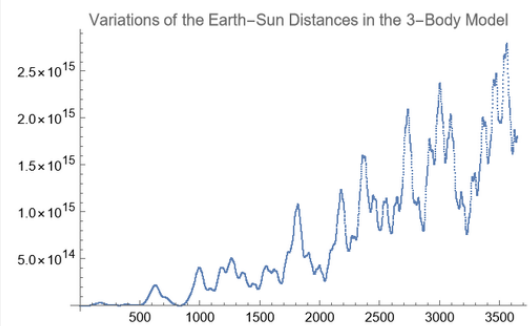
In[347]:=

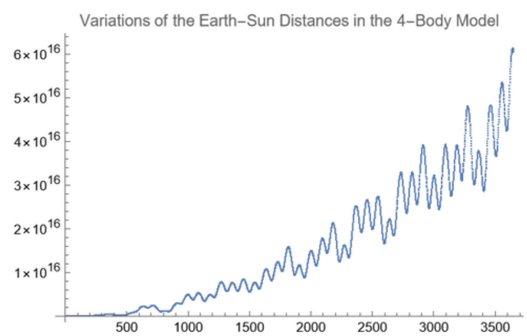
```
ImageResize[GraphicsGrid[
  Partition[Append[MapThread[ListPlot[#1, PlotLabel → #2, ImageSize → {400, 400}] &,
     {{varMoonSun3BodyICRS, varMoonSun4BodyICRS,
       varMoonEarth3BodyICRS, varMoonEarth4BodyICRS,
       varEarthSun2BodyICRS, varEarthSun3BodyICRS, varEarthSun4BodyICRS},
      {"Variations of the Moon-Sun Distances in the 4-Body Model",
       "Variations of the Moon-Sun Distances in the 3-Body Model",
       "Variations of the Moon-Earth Distances in the 3-Body Model",
       "Variations of the Moon-Earth Distances in the 4-Body Model",
       "Variations of the Earth-Sun Distances in the 2-Body Model",
       "Variations of the Earth-Sun Distances in the 3-Body Model",
       "Variations of the Earth-Sun Distances in the 4-Body Model"}}], SpanFromBoth],
   2], Frame → All, FrameStyle → LightGray, ImageSize → {950, 1700}], 1200]
```

Out[*]=









From the plots, we can see that the difference between the models and AstroPosition data get larger with fluctuations over time. Like we said before, this is most likely due to model drift. Notice how the variation increases much faster for the 2-body model than the 3-body model and likewise between the 3-body model to the 4-body model.

It can be seen that all though the deviation from the AstroPosition model increases over time in each model, the Moon-Sun distance graphs have the most scattered data. This is likely because the Moon’s orbit is not exactly elliptical like the Earth’s and is very hard to predict due to the perturbation force by the Sun. This causes the data to scatter more when compared to the Earth-Sun system or the Earth-Moon system.

We can also plot the differences in the distances between models and AstroPosition for different bodies. Below, we define these differences for the distance between the Earth and the Sun and plot them.

In[348]:=

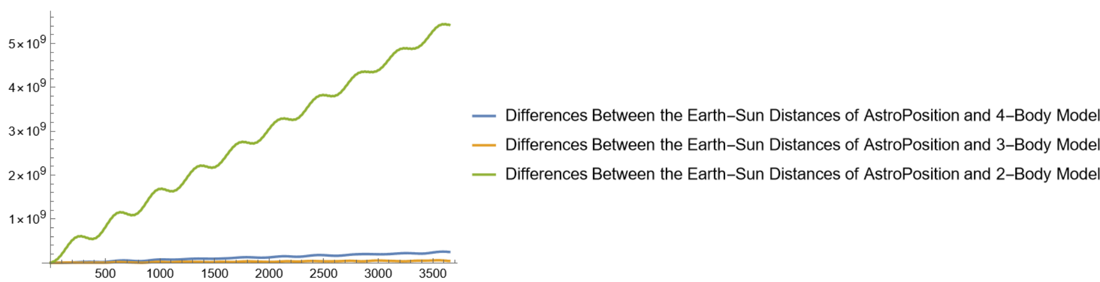
```
differencesEarthSun4BodyandAstroICRS =
  normDifferences[sunEarthSimDistances4BodyICRS, sunEarthDistancesICRS];
differencesEarthSun3BodyandAstroICRS =
  normDifferences[sunEarthSimDistances3BodyICRS, sunEarthDistancesICRS];
differencesEarthSun2BodyandAstroICRS =
  normDifferences[sunEarthSimDistances2BodyICRS, sunEarthDistancesICRS];
differencesEarthSun4and3BodyICRS =
  normDifferences[sunEarthSimDistances4BodyICRS, sunEarthSimDistances3BodyICRS];
```

Once again we will plot them to visualize the different between the Earth-Sun system using the AstroPosition and the models we made .

In[504]:=

```
ListLinePlot[{differencesEarthSun4BodyandAstroICRS,
  differencesEarthSun3BodyandAstroICRS, differences2BodyICRS}, PlotLegends →
  {"Differences Between the Earth–Sun Distances of AstroPosition and 4–Body Model",
   "Differences Between the Earth–Sun Distances of AstroPosition and 3–Body Model",
   "Differences Between the Earth–Sun Distances of AstroPosition and 2–Body Model"}]
```

Out[ ]=



Plotting these differences, we can see the biggest difference is in the 2-body model. However, instead of the 4-body model having the least amount of errors, the 3-body model has it. This is quite different than most of the other results we saw where adding more bodies made the models more accurate. This suggests that adding more bodies may not always create the ideal representation. This may happen because we are looking solely on the Earth-Sun system. In the AstroPosition Data, we also account for the gravitational pull of Mercury and Venus on Earth which counters Jupiter's gravitational pull on the Earth (we can easily verify this using Newton's Law of Gravitation). In fact the difference between the forces on Earth from Jupiter and from Venus and Mercury differ by around 0.2*10^24 N. Hence in the 4-body model, we completely disregard the pull of Venus and Mercury which causes our data to deviate. However in the 3-body model, we ignore both the effects of the Jupiter and the other planets which makes the model more accurate.

We will now compare the differences between the values of the distance between the Moon and the Sun in the models and the AstroPosition model numerically.

In[356]:=

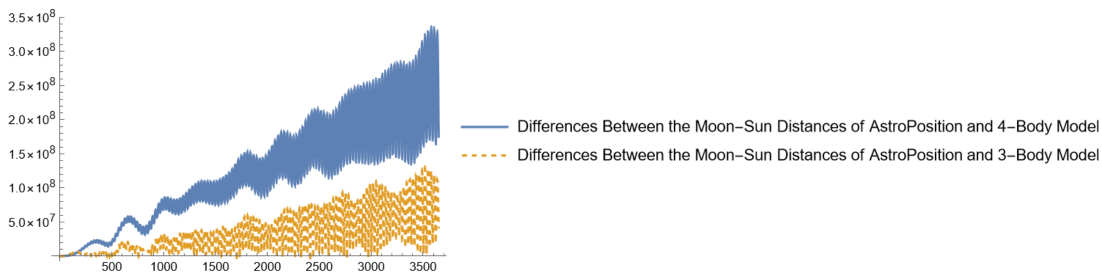
```
differencesMoonSun4BodyandAstroICRS =
  normDifferences[sunMoonSimDistances4BodyICRS, sunMoonDistancesICRS];
differencesMoonSun3BodyandAstroICRS =
  normDifferences[sunMoonSimDistances3BodyICRS, sunMoonDistancesICRS];
```

We will now plot the differences to visually see what is happening.

In[180]:=

```
ListLinePlot[{differencesMoonSun4BodyandAstroICRS,
  differencesMoonSun3BodyandAstroICRS}, PlotLegends →
  {"Differences Between the Moon-Sun Distances of AstroPosition and 4-Body Model",
   "Differences Between the Moon-Sun Distances of AstroPosition and 3-Body Model"},
 PlotStyle → {Default, Dashed}]
```

Out[ ]=



When we plot the difference between the Moon-Sun system with the AstroPosition data and our models, it can be seen that the difference between them get larger with time, and the 4-body model has again more errors but with also more deviations and fluctuations. The reason for this is similar to that explained above.

Repeating the same steps, we get the differences for the distances between the Moon and the Earth.

In[358]:=

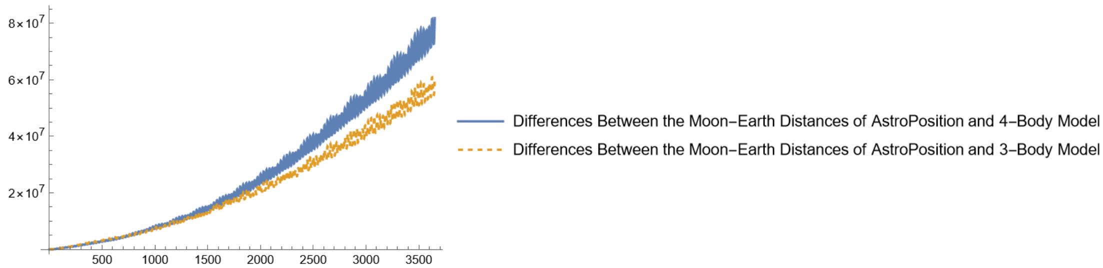
```
differencesMoonEarth4BodyandAstroICRS =
  normDifferences[earthMoonSimDistances4BodyICRS, earthMoonDistancesICRS];
differencesMoonEarth3BodyandAstroICRS =
  normDifferences[earthMoonSimDistances3BodyICRS, earthMoonDistancesICRS];
```

Once again we will plot the differences to get a visual representation of what’s happening.

In[510]:=

```
ListLinePlot[{differencesMoonEarth4BodyandAstroICRS,
  differencesMoonEarth3BodyandAstroICRS}, PlotLegends →
  {"Differences Between the Moon-Earth Distances of AstroPosition and 4-Body Model",
   "Differences Between the Moon-Earth Distances of AstroPosition and 3-Body Model"},
 PlotStyle → {Default, Dashed}]
```

Out[#]=



Like before, the total difference seems to increase with time exponentially, but unlike the other plots, this time the 4-body model has the least errors. As the data that are related to the Sun such as the Earth-Sun distance has the 3-body model as the least erroneous model, the issue might be more related to the Sun than it is to the Moon and the Earth.

We will now take the mean distance difference and turn it into a percentage to create a dataset like we did for the position data.

In[360]:=

```
modelsMeanDistanceDifferences4Body =
  modelsMeanDifferences[{differencesEarthSun4BodyandAstroICRS,
    differencesMoonSun4BodyandAstroICRS, differencesMoonEarth4BodyandAstroICRS}];
modelsMeanDistanceDifferences3Body =
  modelsMeanDifferences[{differencesEarthSun3BodyandAstroICRS,
    differencesMoonSun3BodyandAstroICRS, differencesMoonEarth3BodyandAstroICRS}];
modelsMeanDistanceDifferences2Body =
  modelsMeanDifferences[{differencesEarthSun2BodyandAstroICRS}];
```

After getting the mean absolute error rate, we take the percentage value of the mean error rate, to be able to compare different data types such as the error rate of the distances and positions.

In[363]:=

```
ratiosDistances4Body = ratioDifferences[{differencesEarthSun4BodyandAstroICRS,
    differencesMoonSun4BodyandAstroICRS, differencesMoonEarth4BodyandAstroICRS},
   {normDistancesEarthSunICRS, normDistancesMoonSunICRS, normDistancesMoonEarthICRS}];
ratiosDistances3Body = ratioDifferences[{differencesEarthSun3BodyandAstroICRS,
    differencesMoonSun3BodyandAstroICRS, differencesMoonEarth3BodyandAstroICRS},
   {normDistancesEarthSunICRS, normDistancesMoonSunICRS, normDistancesMoonEarthICRS}];
ratiosDistances2Body =
  ratioDifferences[{differencesEarthSun2BodyandAstroICRS}, {normDistancesEarthSunICRS}];
```

We combine the absolute and percentage error rate to create a data table like before.

In[366]:=

```
differenceRatioList4Body =
  Partition[Riffle[modelsMeanDistanceDifferences4Body, ratiosDistances4Body], 2];
differenceRatioList3Body =
  Partition[Riffle[modelsMeanDistanceDifferences3Body, ratiosDistances3Body], 2];
differenceRatioList2Body =
  Partition[Riffle[modelsMeanDistanceDifferences2Body, ratiosDistances2Body], 2];
```

In[369]:=

```
dsDistances = Dataset[{Association["4-Body" →
      AssociationThread[{"Distance Between Earth-Sun", "Distance Between Moon-Sun",
         "Distance Between Moon-Earth"} → differenceRatioList4Body]],
    Association["3-Body" → AssociationThread[{"Distance Between Earth-Sun",
         "Distance Between Moon-Sun", "Distance Between Moon-Earth"} →
        differenceRatioList3Body]], Association["2-Body" →
      AssociationThread[{"Distance Between Earth-Sun"} → differenceRatioList2Body]]}]
```

Out[∘]=

4–Body	Distance Between Earth–Sun	$\{1.19667\times10^{8}\,\mathrm{m},\ 0.0799928\%\}$
	Distance Between Moon–Sun	$\{1.20741\times10^{8}\,\mathrm{m},\ 0.0807293\%\}$
	Distance Between Moon–Earth	$\{2.22382\times10^{7}\,\mathrm{m},\ 5.79446\%\}$
3–Body	Distance Between Earth–Sun	$\{2.58442\times10^{7}\,\mathrm{m},\ 0.0172688\%\}$
	Distance Between Moon–Sun	$\{3.78899\times10^{7}\,\mathrm{m},\ 0.0253418\%\}$
	Distance Between Moon–Earth	$\{3.14926\times10^{7}\,\mathrm{m},\ 8.20451\%\}$
2–Body	Distance Between Earth–Sun	$\{2.77203\times10^{9}\,\mathrm{m},\ 1.85257\%\}$

As explained before, the 4-body model has the least error when it comes to the distance between the Earth and the Moon, while for other distances the 3-body model has the least error. However, all of them are very small percentages with the largest one around 1/125 of the AstroPosition value at most.

Comparison of Velocities With the Data from AstroPosition and Different Models

After comparing the distances and positions of the bodies, we will now compare the velocities of the bodies. As the velocities from the simulation and velocities of the Earth and the Sun from AstroPosition are already defined, we only need to define the velocity of the Moon from AstroPosition over time here.

In[370]:=

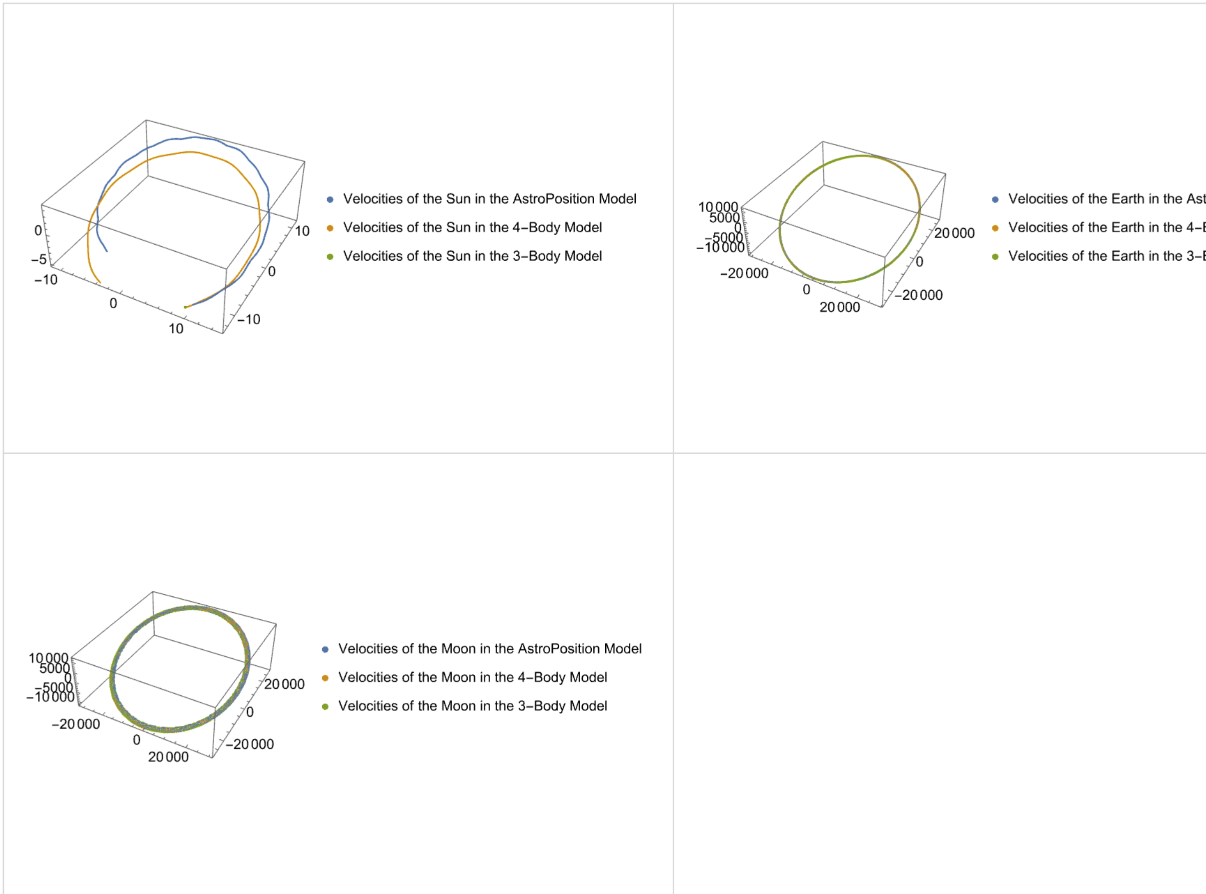
```
moonVelocitiesICRS =
  (Table[positionICRS[moon, DatePlus[i, Quantity[0.5, "Minutes"]]], {i, dates}] -
    moonPositionsICRS) / UnitConvert[0.5 min, "Seconds"];
```

When we plot the velocities in 3D, the data for the velocities of the Earth and the Moon seem to overlap, while the deviations for the data of the velocity of the Sun are more apparent.

In[458]:=

```
GraphicsGrid[Partition[Append[MapThread[ListPointPlot3D[#1,
      PlotLegends → {"Velocities of the " <> #2 <> " in the AstroPosition Model",
        "Velocities of the " <> #2 <> " in the 4-Body Model",
        "Velocities of the " <> #2 <> " in the 3-Body Model"}, ImageSize → {200, 300}] &,
    {{{sunVelocitiesICRS, sunSimVel4BodyICRS, sunSimVel3BodyICRS},
      {earthVelocitiesICRS, earthSimVel4BodyICRS, earthSimVel3BodyICRS},
      {moonVelocitiesICRS, moonSimVel4BodyICRS, moonSimVel3BodyICRS}},
     {"Sun", "Earth", "Moon"}}], ""], 2],
 Frame → All, FrameStyle → LightGray, ImageSize →
  {1000, 700}]
```

Out[ ]=



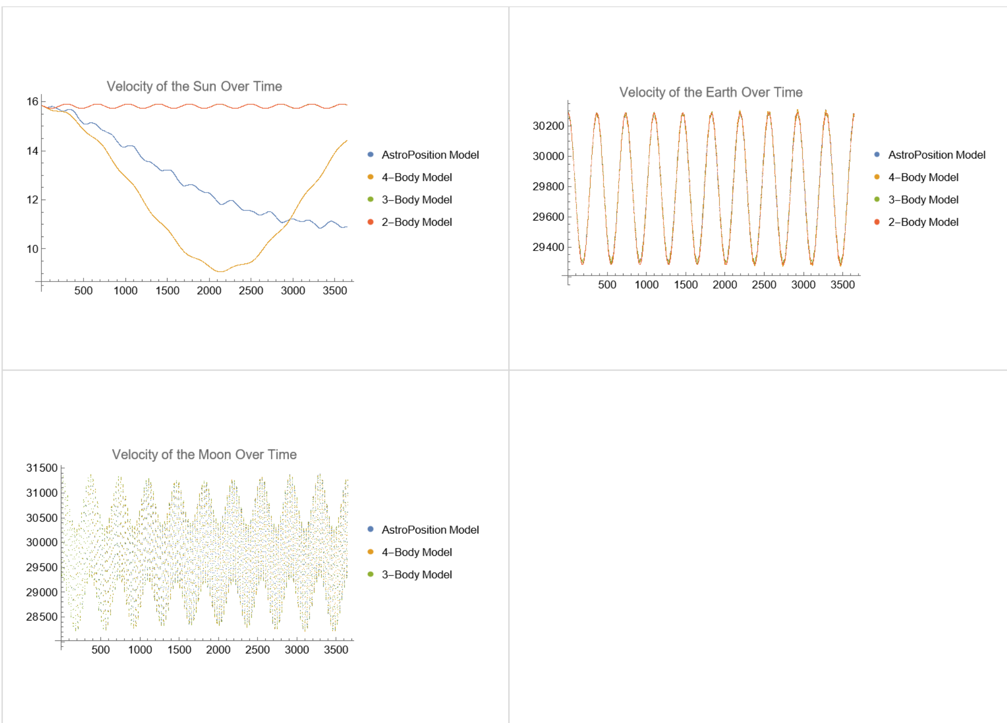
It is quite interesting that Moon’s velocities are is different patterns than that of the Earth and the Sun. This could happen because of the Moon’s complex orbit which makes the velocity hard to describe in one analytic formula.

Now we will graph the comparisons of the difference models with the velocities.

In[393]:=

```
GraphicsGrid[Partition[Append[
   MapThread[ListPlot[#1, PlotLegends → Which[Length[#1] == 4, {"AstroPosition Model",
          "4-Body Model", "3-Body Model", "2-Body Model"}, Length[#1] == 3,
         {"AstroPosition Model", "4-Body Model", "3-Body Model"}, True, ""],
      PlotLabel → "Velocity of the " <> #2 <> " Over Time",
      PlotRange → All, ImageSize → {300, 300}] &,
    {{{normData[sunVelocitiesICRS], normData[sunSimVel4BodyICRS],
       normData[sunSimVel3BodyICRS], normData[sunSimVel2BodyICRS]},
      {normData[earthVelocitiesICRS], normData[earthSimVel4BodyICRS],
       normData[earthSimVel3BodyICRS], normData[earthSimVel2BodyICRS]},
      {normData[moonVelocitiesICRS], normData[moonSimVel4BodyICRS],
       normData[moonSimVel3BodyICRS]}}, {"Sun", "Earth", "Moon"}}], ""], 2],
 Frame → All, FrameStyle → LightGray, ImageSize →
  {1000, 700}]
```

Out[ ]=



Plotting the norms of the velocities in 2D shows that the velocity of the Sun in the 4-body model is closer to the data from AstroPosition than the velocities in the 2 and 3-body model are. However, the velocities of the Moon and the Earth seem to overlap. Thus, we will again investigate them using a numerical approach.

We take the differences between the models and AstroPosition model and apply the same steps we used before to get the percentage error. First, we take the differences of the velocities of the Sun, Earth and Moon separately.

In[394]:=

```
sunVelDifferences = Map[normDifferences[#, sunVelocitiesICRS] &,
   {sunSimVel4BodyICRS, sunSimVel3BodyICRS, sunSimVel2BodyICRS}];
earthVelDifferences = Map[normDifferences[#, earthVelocitiesICRS] &,
   {earthSimVel4BodyICRS, earthSimVel3BodyICRS, earthSimVel2BodyICRS}];
moonVelDifferences = Map[normDifferences[#, moonVelocitiesICRS] &,
   {earthSimVel3BodyICRS, earthSimVel2BodyICRS}];
```

Now, we separate the lists above according to models.

In[397]:=

```
differencesSunVel4BodyandAstroICRS = sunVelDifferences⟦1⟧;
differencesSunVel3BodyandAstroICRS = sunVelDifferences⟦2⟧;
differencesSunVel2BodyandAstroICRS = sunVelDifferences⟦3⟧;
differencesEarthVel4BodyandAstroICRS = earthVelDifferences⟦1⟧;
differencesEarthVel3BodyandAstroICRS = earthVelDifferences⟦2⟧;
differencesEarthVel2BodyandAstroICRS = earthVelDifferences⟦3⟧;
differencesMoonVel4BodyandAstroICRS = moonVelDifferences⟦1⟧;
differencesMoonVel3BodyandAstroICRS = moonVelDifferences⟦2⟧;
```

Here, we take the average value of each list, giving us the average absolute error rate.

In[405]:=

```
modelsMeanVelDifferences4Body =
  modelsMeanDifferences[{differencesSunVel4BodyandAstroICRS,
    differencesEarthVel4BodyandAstroICRS, differencesMoonVel4BodyandAstroICRS}];
modelsMeanVelDifferences3Body =
  modelsMeanDifferences[{differencesSunVel3BodyandAstroICRS,
    differencesEarthVel3BodyandAstroICRS, differencesMoonVel3BodyandAstroICRS}];
modelsMeanVelDifferences2Body = modelsMeanDifferences[
   {differencesSunVel2BodyandAstroICRS, differencesEarthVel2BodyandAstroICRS}];
```

We use these lists again to calculate the percentage error this time.

In[408]:=

```
ratiosVel4Body = ratioDifferences[
   {differencesSunVel4BodyandAstroICRS, differencesEarthVel4BodyandAstroICRS,
    differencesMoonVel4BodyandAstroICRS}, {normData[sunVelocitiesICRS],
    normData[earthVelocitiesICRS], normData[moonVelocitiesICRS]}];
ratiosVel3Body = ratioDifferences[
   {differencesSunVel3BodyandAstroICRS, differencesEarthVel3BodyandAstroICRS,
    differencesMoonVel3BodyandAstroICRS}, {normData[sunVelocitiesICRS],
    normData[earthVelocitiesICRS], normData[moonVelocitiesICRS]}];
ratiosVel2Body = ratioDifferences[
   {differencesSunVel2BodyandAstroICRS, differencesEarthVel2BodyandAstroICRS},
   {normData[sunVelocitiesICRS], normData[earthVelocitiesICRS]}];
```

In[411]:=
```
differenceVelRatioList4Body =
  Partition[Riffle[modelsMeanVelDifferences4Body, ratiosVel4Body], 2];
differenceVelRatioList3Body =
  Partition[Riffle[modelsMeanVelDifferences3Body, ratiosVel3Body], 2];
differenceVelRatioList2Body =
  Partition[Riffle[modelsMeanVelDifferences2Body, ratiosVel2Body], 2];
```

Now we will create a data table to view the differences and percent errors.

In[414]:=
```
dsVel =
 Dataset[{Association["4-Body" → AssociationThread[{"Sun Velocity", "Earth Velocity",
        "Moon Velocity"} → differenceVelRatioList4Body]], Association[
    "3-Body" → AssociationThread[{"Sun Velocity", "Earth Velocity", "Moon Velocity"} →
       differenceVelRatioList3Body]], Association["2-Body" → AssociationThread[
      {"Sun Velocity", "Earth Velocity"} → differenceVelRatioList2Body]]}]
```

Out[ ]=

4–Body	Sun Velocity	{2.71478 m/s, 22.7506%}
	Earth Velocity	{24.0399 m/s, 0.0807874%}
	Moon Velocity	{1022.2 m/s, 3.434%}
3–Body	Sun Velocity	{20.0946 m/s, 163.349%}
	Earth Velocity	{20.6703 m/s, 0.0695266%}
	Moon Velocity	{1114.43 m/s, 3.74334%}
2–Body	Sun Velocity	{20.0947 m/s, 163.35%}
	Earth Velocity	{552.105 m/s, 1.85439%}

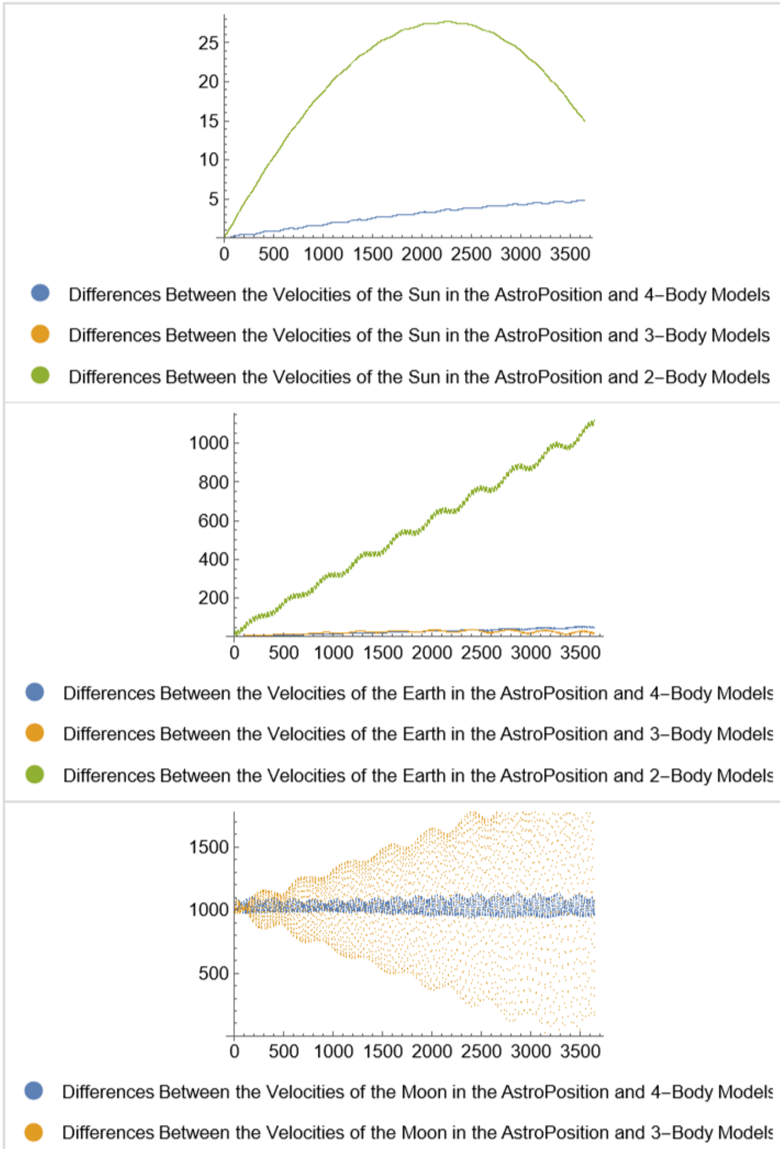
We can see that the 4-body model has the least difference for the Sun and the Moon but the 3-body model has the least error for the velocity of the Earth, even though the two models (4 and 3-body models) have very close values with only 0.02% difference. The biggest difference across models is the difference between the 3 and 4-body model with the value decreasing nearly to 1/125 of its value.

Like we did before we will now compare the models we built with the AstroPosition Data to get a visual representation.

In[449]:=

```
GraphicsColumn[MapThread[ListPlot[#1,
    PlotLegends → Which[Length[#1] == 3, {"Differences Between the Velocities of the " <>
        #2 <> " in the AstroPosition and 4-Body Models",
       "Differences Between the Velocities of the " <> #2 <>
        " in the AstroPosition and 3-Body Models",
       "Differences Between the Velocities of the " <> #2 <>
        " in the AstroPosition and 2-Body Models"},
      Length[#1] == 2, {"Differences Between the Velocities of the " <>
        #2 <> " in the AstroPosition and 4-Body Models",
       "Differences Between the Velocities of the " <> #2 <>
        " in the AstroPosition and 3-Body Models"}, True, ""],
    PlotStyle → {Default, Dashed}, ImageSize → {230, 140}] &,
  {{{differencesSunVel4BodyandAstroICRS, differencesSunVel3BodyandAstroICRS,
     differencesSunVel2BodyandAstroICRS},
    {differencesEarthVel4BodyandAstroICRS, differencesEarthVel3BodyandAstroICRS,
     differencesEarthVel2BodyandAstroICRS},
    {differencesMoonVel4BodyandAstroICRS, differencesMoonVel3BodyandAstroICRS}},
   {"Sun", "Earth", "Moon"}}],
 Frame → All, FrameStyle → LightGray,
 ImageSize →
  {600, 650}]
```

Out[ ]=



When we plot the differences in the velocities of 2 and 3-body models, they seem to overlap for the Sun as the difference is a factor of 10^-4. For other velocity differences, however, it seems to change which model has the least error. As the data table was built by using mean values, it’s not possible to see the changes over time. But with these plots, it can be seen that. For example, the difference between the 2-body model and AstroPosition data for the Earth decreases after a certain point while the difference between the 4-body and AstroPosition model increases with time for our date range.

Another interesting graph here is the the difference of the velocities of the Moon between the 3-Body model and the 4-Body model. We can see that the differences of velocities of the Moon in the 4-Body system is more stable and centered around 1000 m/s while the difference of velocities of the Moon in 3-Body system disperses and diverges. Once again this is most likely due to model drifting.

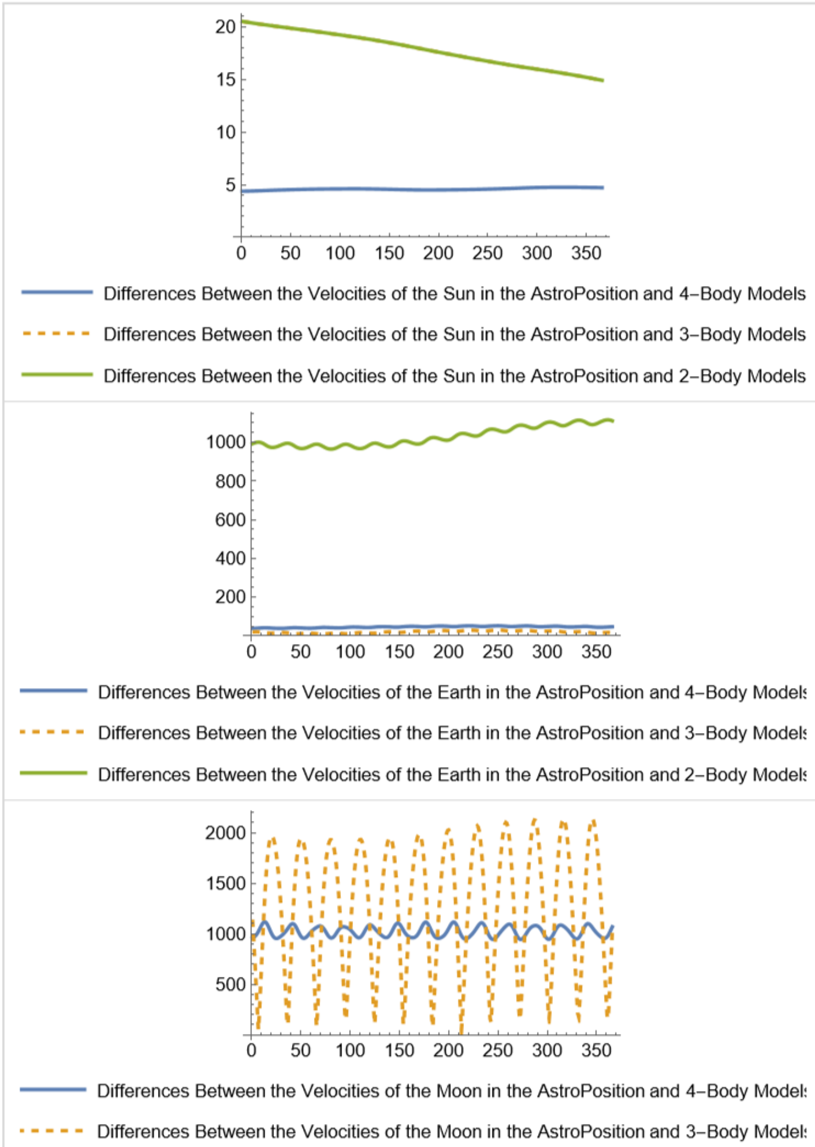
However, many of the plots above contain too many close data points, it’s hard to make sense of the velocity differences for the Moon. Hence, we will take a closer look by focusing on the last year.

In[434]:=
```
differencesSunVel4BodyandAstroICRSLastYear =
  lastYear[differencesSunVel4BodyandAstroICRS];
differencesSunVel3BodyandAstroICRSLastYear =
  lastYear[differencesSunVel3BodyandAstroICRS];
differencesSunVel2BodyandAstroICRSLastYear =
  lastYear[differencesSunVel2BodyandAstroICRS];
differencesEarthVel4BodyandAstroICRSLastYear =
  lastYear[differencesEarthVel4BodyandAstroICRS];
differencesEarthVel3BodyandAstroICRSLastYear =
  lastYear[differencesEarthVel3BodyandAstroICRS];
differencesEarthVel2BodyandAstroICRSLastYear =
  lastYear[differencesEarthVel2BodyandAstroICRS];
differencesMoonVel4BodyandAstroICRSLastYear =
  lastYear[differencesMoonVel4BodyandAstroICRS];
differencesMoonVel3BodyandAstroICRSLastYear =
  lastYear[differencesMoonVel3BodyandAstroICRS];
```

Once again we will draw out the visual representations of the comparisons of the model.

In[448]:=
```
GraphicsColumn[MapThread[ListLinePlot[#1,
    PlotLegends → Which[Length[#1] == 3, {"Differences Between the Velocities of the " <>
         #2 <> " in the AstroPosition and 4-Body Models",
        "Differences Between the Velocities of the " <> #2 <>
         " in the AstroPosition and 3-Body Models",
        "Differences Between the Velocities of the " <> #2 <>
         " in the AstroPosition and 2-Body Models"},
       Length[#1] == 2, {"Differences Between the Velocities of the " <>
         #2 <> " in the AstroPosition and 4-Body Models",
        "Differences Between the Velocities of the " <> #2 <>
         " in the AstroPosition and 3-Body Models"}, True, ""],
    PlotStyle → {Default, Dashed}, ImageSize → {230, 140}] &,
  {{{differencesSunVel4BodyandAstroICRSLastYear,
     differencesSunVel3BodyandAstroICRSLastYear,
     differencesSunVel2BodyandAstroICRSLastYear},
    {differencesEarthVel4BodyandAstroICRSLastYear,
     differencesEarthVel3BodyandAstroICRSLastYear,
     differencesEarthVel2BodyandAstroICRSLastYear},
    {differencesMoonVel4BodyandAstroICRSLastYear,
     differencesMoonVel3BodyandAstroICRSLastYear}}, {"Sun", "Earth", "Moon"}}],
 Frame → All, FrameStyle → LightGray, ImageSize →
  {600, 650}]
```

Out[•]=



From the plots below, it can be seen that the 2-body simulation for the Moon has more errors than the 3-body simulation. Similarly, for the differences in the Earth's velocity, the biggest errors are seen in the 2-body simulation and the smallest errors in 4-body simulation which suggests adding the Jupiter made the simulation closer to the data from AstroPosition for the prediction of the data of the Sun and Moon.

Conclusion & Further Research

In this project, we looked at how well 2-body, 3-body and 4-body model predict the movements of the Earth, Moon and Sun, using data from the AstroPosition function with the ICRS coordinate system for comparison. We considered several perspectives for this comparison: the positions, velocities of the bodies and the distances between each one.

The data tables below show that adding more bodies to the model generally increases the accuracy of predictions. While the 2-body model is simple, it has larger errors because it doesn't consider other factors like the Moon's gravity. The 3-body model, which adds the Moon, doesn't change the accuracy much for the velocities and positions that much while decreasing the error in the distances between the Earth and Sun to nearly 0.0091 of the error in the 2-body model. Moreover, the 4-body model with Jupiter helps reduce errors again for most of the data but increases errors for the distances between the Earth and Sun and the velocity of the Earth which shows that more complex models don't always give better results for all aspects, but they help increase accuracy for certain parts of the system and that even though the Jupiter is far away, its gravity affects the Sun's movement. However, the project is limited to only 4 bodies; considering other celestial influences could further improve accuracy. Future research could include more bodies, plots, calculations or different coordinate systems.

In[455]:=

```
Column[{dsPositions, dsDistances, dsVel}]
```

Out[•]=

4–Body	Position of the Sun	$\{2.91499\times10^{8}$ m, 40.7465%$\}$
	Position of the Earth	$\{3.15636\times10^{8}$ m, 0.210685%$\}$
	Position of the Moon	$\{3.15951\times10^{8}$ m, 0.210903%$\}$
3–Body	Position of the Sun	$\{2.51275\times10^{9}$ m, 352.62%$\}$
	Position of the Earth	$\{2.51384\times10^{9}$ m, 1.68051%$\}$
	Position of the Moon	$\{2.51385\times10^{9}$ m, 1.68052%$\}$
2–Body	Position of the Sun	$\{2.51274\times10^{9}$ m, 352.619%$\}$
	Position of the Earth	$\{3.52761\times10^{9}$ m, 2.35555%$\}$

4–Body	Distance Between Earth–Sun	$\{1.19667\times10^{8}$ m, 0.0799928%$\}$
	Distance Between Moon–Sun	$\{1.20741\times10^{8}$ m, 0.0807293%$\}$
	Distance Between Moon–Earth	$\{2.22382\times10^{7}$ m, 5.79446%$\}$
3–Body	Distance Between Earth–Sun	$\{2.58442\times10^{7}$ m, 0.0172688%$\}$
	Distance Between Moon–Sun	$\{3.78899\times10^{7}$ m, 0.0253418%$\}$
	Distance Between Moon–Earth	$\{3.14926\times10^{7}$ m, 8.20451%$\}$
2–Body	Distance Between Earth–Sun	$\{2.77203\times10^{9}$ m, 1.85257%$\}$

4–Body	Sun Velocity	{2.71478 m/s, 22.7506%}
	Earth Velocity	{24.0399 m/s, 0.0807874%}
	Moon Velocity	{1022.2 m/s, 3.434%}
3–Body	Sun Velocity	{20.0946 m/s, 163.349%}
	Earth Velocity	{20.6703 m/s, 0.0695266%}
	Moon Velocity	{1114.43 m/s, 3.74334%}
2–Body	Sun Velocity	{20.0947 m/s, 163.35%}
	Earth Velocity	{552.105 m/s, 1.85439%}

We also would like to expand the project to more than just four bodies. For example how would the analysis change if we had a binary star system or multiple moons. Many of these challenges require a more in-depth study that we wish to pursue in the future.

References

1. NASA. (n.d.). Eclipses and the Moon’s orbit. NASA. https://eclipse.gsfc.nasa.gov/SEhelp/moonorbit.html

2. Goldstein, Bernard R. “Ancient and Medieval Values for the Mean Synodic Month” University of Pittsburgh, https://sites.pitt.edu/~brg/pdfs/brg_i_2.pdf. Accessed 29 Jan. 2025.

3. Wolfram, Stephen. “When Exactly Will the Eclipse Happen? A Multimillennium Tale of Computation.” Stephen Wolfram Writings, 27 Mar. 2024, https://writings.stephenwolfram.com/2024/03/when-exactly-will-the-eclipse-happen-a-multimillennium-tale-of-computation/. Accessed 29 Jan. 2025.

4. Wolfram, Steven. “When Exactly Will the Eclipse Happen? A Multimillennium Tale of Computation ” YouTube, March 2024, https://www.youtube.com/watch?v=7Eqhd34ytoc

Acknowledgements

We would like to thank our mentor Brian Silverman for countless hours of help and guidance on our research and our weekly meetings to learn about ancient civilizations and their calendar systems and interesting astronomy factoids. We would also like to thank Daniel Sanchez for his feedback on our project and for offering new ideas, snippets of code when we struggled, and helping navigate problems with AstroPosition. We would also like to thank Stephen Wolfram and the directors (Rory Foulger, Eryn Gillman) for organizing the program and offering us a chance to work with amazing people from around the world.

Lastly we would like to thank our families for their support and encouragement through the six months of the project.