

Chapter 4

The Numerical Algorithm to Find the Halo Orbits

The method we use to find the halo orbits is to start with a certain family of small orbits around L_2 , which can be found by using the linearized vector field, and then to continue this family to larger and larger sizes. We look for a point where two other families branch off from this one; this point corresponds to two eigenvalues of the Poincaré map colliding at $+1$ and moving off of the unit circle.

We have numerically integrated both the Hamiltonian vector field X_H (*i.e.* integrated with the co-ordinates (x, y, z, p_x, p_y, p_z)) and the Lagrangian vector field X (co-ordinates $(x, y, z, \dot{x}, \dot{y}, \dot{z})$). The two approaches are quite similar. Indeed, as mentioned in Chapter 1, they share the same configuration space and the only difference is $p_x = \dot{x} - y$ and $p_y = \dot{y} + x$. In this chapter, some discussions are more easily phrased in terms of X_H and others in terms of X . Of course, in the event of any confusion, it is always possible to transform one formulation into the other using the Legendre transformation.

4.1 Using Linearized Vector Field to Approximate Small Orbits

The linearization of a vector field $Y : \mathbb{R}^N \rightarrow \mathbb{R}^N$ about a point y_0 is defined to be DY_{y_0} ; it is taking the first order term from the Taylor expansion

of Y and truncating all the higher order terms. The vector field we want to work with is the Lagrangian vector field X that was introduced in Chapter 1. $DX_{(x,y,z,\dot{x},\dot{y},\dot{z})}$ is explicitly exhibited on p. 29. Evaluating it at the point L_2 yields¹

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & & \\ 0 & 0 & 0 & 1 & & \\ (1+2\Delta) & 0 & 0 & -2 & & \\ 0 & (1-\Delta) & 2 & 0 & & \\ & & & & 0 & 1 \\ & & & & -\Delta & 0 \end{bmatrix}$$

where $\Delta = \frac{1-\mu}{r_1^3(L_2)} + \frac{\mu}{r_2^3(L_2)}$. In Chapter 2, the upper 4x4 block was called B , and the lower 2x2 C . We take the point L_2 as our origin, and we want to solve the linear equation $\dot{u} = Au$. The general solution is, of course, $u(t) = e^{At}u_0$, where u_0 is the vector of initial conditions. To compute e^{At} , A is reduced to its Jordan normal form $A = Q^{-1}JQ$. The J in this discussion should not be confused with the standard symplectic matrix J . Converting to Jordan form involves making the change of variables $w = Qu$, and the linear differential equation then becomes $\dot{w} = Jw$, and $e^{At} = Q^{-1}e^{Jt}Q$.

Looking in Szebehely [1967 p.217], the numerical values for the Earth-Moon system are

$$\begin{aligned} L_2 &= (1.1556824834, 0, 0) \\ \mu &= 0.0121506683 \end{aligned}$$

We then calculate $\Delta = 3.190423608$. Using this value, and referring to our

¹We recall that the variables were reordered as $(x, y, \dot{x}, \dot{y}, z, \dot{z})$ to bring out the block structure more clearly.

related work in Chapter 2, we can explicitly calculate $\alpha = 1.8626454$ and $\beta = 1.7861757$.

Remark *This is the only part of the algorithm described in this chapter that is particular to the Earth-Moon system; by changing μ and computing the coordinates of the corresponding point L_2 , this class of halo orbits can be computed for other R3BP systems, such as the Sun-Earth.*

We need to explicitly compute J and Q^{-1} in order to be able to choose initial conditions to make a periodic orbit. This could have easily been done with the EISPACK routine RG, as was done later on in the HALO program, but to get started, the author used MAPLE. The matrix J was found to be diagonal, and so e^{Jt} is simply the exponential of each of the diagonal terms. $w(t) = e^{Jt}w_0$ is, displaying only 3 digits of accuracy here (all that is needed is a crude approximation),

$$w(t) = \begin{bmatrix} e^{-i(1.78t)} & 0 & 0 & 0 & 0 & 0 \\ 0 & e^{i(1.78t)} & 0 & 0 & 0 & 0 \\ 0 & 0 & e^{-2.15t} & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{2.15t} & 0 & 0 \\ 0 & 0 & 0 & 0 & e^{-i(1.86t)} & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{i(1.86t)} \end{bmatrix} \begin{bmatrix} w_0^1 \\ w_0^2 \\ w_0^3 \\ w_0^4 \\ w_0^5 \\ w_0^6 \end{bmatrix}$$

It is obvious we must take $w_0^3 = 0 = w_0^4$ to kill the nonperiodic terms. Then since $u_0 = Q^{-1}w_0$, we must take $w_0^1 = w_0^2 := \gamma$ and $w_0^5 = w_0^6 := \delta$ to kill the imaginary components, as can be seen from

$$\begin{bmatrix} u_0^1 \\ u_0^2 \\ u_0^3 \\ u_0^4 \\ u_0^5 \\ u_0^6 \end{bmatrix} = \begin{bmatrix} 0 & 0 & .667 & .667 & -.167 & -.167 \\ 0 & 0 & -1.44 & 1.44 & .311i & -.311i \\ 0 & 0 & .420 & -.420 & .487i & -.487i \\ 0 & 0 & -.907 & -.907 & .907 & .907 \\ .500 & .500 & 0 & 0 & 0 & 0 \\ -.893i & .893i & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_0^1 \\ w_0^2 \\ 0 \\ 0 \\ w_0^5 \\ w_0^6 \end{bmatrix}$$

Our vector of periodic initial conditions is then

$$[x_0, y_0, \dot{x}_0, \dot{y}_0, z_0, \dot{z}_0]^T = u_0^T = [-.334\delta, 0, 0, 1.814\delta, \gamma, 0]^T \quad (4.1)$$

and the general periodic solution looks like

$$u(t) = \begin{bmatrix} -\delta(0.334) \cos(1.86t) \\ \delta(0.622) \sin(1.86t) \\ \delta(0.956) \sin(1.86t) \\ \delta(1.814) \cos(1.86t) \\ \gamma(1.000) \cos(1.78t) \\ \gamma(1.786) \sin(1.78t) \end{bmatrix}$$

If $\gamma \neq 0$, $\delta = 0$, then a periodic orbit is a straight line oscillation along the z -axis. If $\delta \neq 0$, $\gamma = 0$, then there are elliptical shaped orbits lying in the (x, y) plane. These are the families of orbits that were shown to persist for the nonlinear vector field by our discussion of Lyapunov's Center theorem in Chapter 2. If both $\gamma \neq 0$, $\delta \neq 0$, then we can only get a periodic orbit if the frequencies of the two separate modes of oscillation are rationally related, *i.e.* $\beta = \frac{p}{q}\alpha$. In our case, this does not happen, so there are no mixed mode oscillations in the linear system. What we want to do is numerically continue the planar family in the full nonlinear problem and search for a point of bifurcation; that is, to find our first set of initial conditions, we put $\gamma = 0$ and take δ small. Intuitively, what is happening is that in the nonlinear situation the frequencies depend upon the size of the orbits, and they become commensurable when the orbits reach a certain size.

4.2 The Algorithm to Refine Initial Conditions

Once the initial conditions that would cause the trajectory of a particle to be a periodic orbit for the linearized vector field are obtained, we attempt

to use this information to find a periodic orbit of the original, nonlinear vector field. There is no nice solution for the R3BP (in terms of special functions or a convergent power series expansion), like there is for, say, the harmonic oscillator; therefore numerical integration is used. We numerically integrate the trajectory of a particle with the same initial conditions that produced a closed orbit in the linearized system. The resulting trajectory is almost a periodic orbit, but it will not close, in general. Here a Newton method is used to iteratively refine the initial conditions until the orbit closes to within a prescribed tolerance. We must be a little careful, however.

The first thought would be to search for zeroes of the function $G(x) := \phi(t, x) - x$ where, ϕ is the flow² of the vector field. This idea is immediately doomed to failure because, a priori, there is not one isolated zero of the function for the Newton method to converge to – there is an entire curve of zeroes, consisting of all points on the periodic orbit that we are hoping to find. Therefore, the second idea is to fix a Poincaré section S and look for fixed points of the Poincaré map, or more precisely, zeroes of the map $G(x) := \Theta(x) - x$ for $x \in S$. Unfortunately, this will also fail, because there is a one parameter family of orbits, and therefore, a curve of zeroes of our map in S . We must reduce our degrees of freedom one step more.

In theoretical discussions, this is done by restricting our Poincaré map to a fixed energy submanifold. Indeed, we shall do this for calculating some

²Note that we have changed our notation for the flow from $F(x, t)$ to $\phi(t, x)$; this will avoid problems later on.

things.³ However, it has been found to be computationally quite effective to fix one of the co-ordinates, say x_1 , and then do the Newton iteration. In this way, we can find the initial conditions for an orbit with a given value of x_1 ; then by incrementing x_1 , we can iterate again and find another orbit. Before we discuss this in detail, let us explain how to numerically compute the Poincaré map Θ . The idea of the method is quite standard, although not easy to find explicitly written out in the literature. After extensive searching, the author found a good discussion in Simó [1989], which is reworked below; Howell [1983] also covered some of the same ground.

4.2.1 The Variational Equations

To use Newton's method, the map $D\Theta_x$ is needed. This is obtained from $D(\phi_T)_x$, the differential of the time T map, where T is the first return time of the flow through the point x . We can compute this numerically by integrating the first order variational equations. First, however, let us define what the variational equations are. A solution to the ordinary differential equation $\dot{\mathbf{x}} = \mathbf{X}(\mathbf{x})$ is a function $\phi(t, x)$ where $x \in \mathbb{R}^6$ is the initial point a particular solution curve $\phi_x(t)$ must pass through at time $t = 0$. In other words

$$\frac{d}{dt}\phi(t, x) = \mathbf{X}(\phi(t, x)) \quad (4.2)$$

for all $x \in \mathbb{R}^6$ and all t in an interval, which depends on x . Let us differentiate both sides with respect to the initial condition point x (i.e. the differential

³See the section of this chapter referring to the failure of Newton's method restricted to a fixed energy surface.

operator D_2 is applied to both sides). On the left hand side, D_2 and $\frac{d}{dt}$ can be interchanged because there is have sufficient smoothness (the vector field for the restricted 3 body problem is C^ω at all points except the locations of the two primaries). Then we obtain

$$\frac{d}{dt}[D_2\phi](t, x) = DX_{\phi(t, x)}D_2\phi(t, x) \quad (4.3)$$

This is a 6×6 matrix differential equation, which can be written as

$$\dot{F} = MF \quad (4.4)$$

Note that M depends on t . The 36 component equations of the above system are called the (linear) variational equations. Higher order variational equations also exist, but we shall not discuss them. If this set of equations is numerically integrated, along with our equations of motion, starting with the initial condition $F(0) = I$ (the 6×6 identity matrix) and the integration is carried out to time T , then $F(T) = D(\phi_T)$. This is just what we wanted.

The numerical integration is carried out using a Runge-Kutta-Fehlberg method of order 8 with 7th order step-size corrections. The specific package was named EDO and was written by Carles Simó; the RK78 subroutine family was used. Of course, all computations were carried out in double precision arithmetic.

Verifying Correctness of Variational Equations The programming of mathematical formulae is extremely prone to error, and therefore we must be careful that the thing we want to calculate is what the computer are actually calculating.

The first check is that the Jacobi constant is preserved by the numerical integration of a trajectory. The value of the Jacobi constant C was computed along periodic orbits, and found typically to remain constant to 15 digits. Of course, knowing that the numerically computed trajectory is confined to a submanifold of constant C is not a guarantee that the trajectory is following the true flow of the vector field; the true flow is also restricted to that submanifold, but the flow could be behaving differently on it. To really pin down the motion in the 3 dimensional restricted problem, two more independent constants of motion would be needed, and they are believed not to exist.

An easy way to look for errors in the coding of the variational equations is to check that the monodromy matrix has determinant equal to 1; we know this must happen because the flow of a Hamiltonian vector field is volume preserving. Here we are integrating the corresponding Lagrangian vector field, but in view of the Legendre transformation discussed earlier, this property carries over. Also the eigenvalues are the same for the monodromy map in both the Hamiltonian and Lagrangian formulations; since the flow of a Hamiltonian vector field is symplectic, the eigenvalues must obey the familiar relations given in Abraham & Marsden [1985 p.168]. Of course, before we even integrate we know that DX_H is an infinitesimally symplectic map and therefore its eigenvalues, and thus those of DX , must obey the infinitesimally symplectic eigenvalue relations.

We can then use the Eigenvalue Inheritance theorem, discussed in Chapter 2, and see if the eigenvalues of the Poincaré map restricted to an

energy submanifold, are indeed a subset of the eigenvalues of the monodromy map. Note carefully that this relationship need *not* hold for the Poincaré map restricted to anything other than a constant *energy* submanifold; not realizing this caused the author many delays.

However useful the above tests are, it is still quite possible for errors to slip by, and indeed this happened. Once, when working with X_H , an error in the formula for 2 partial derivatives in the symplectic variational equations went hidden for a long time because the resulting monodromy matrix still managed to be symplectic.

Another way to check the variational equations is to fix a time value t (preferably small), a point $x_0 \in \mathbb{R}^{2n}$, and a vector $v \in \mathbb{R}^{2n}$. We then want to check that, for $h \in \mathbb{R}$,

$$\| \phi_t(x_0 + hv) - (\phi_t(x_0) + D(\phi_t)_{x_0}(hv)) \| = \mathcal{O}(h^2)$$

The idea then is to numerically compute $\phi_t(x_0 + hv)$ and $[\phi_t(x_0) + D(\phi_t)_{x_0}(hv)]$ for a decreasing sequence of values of h , and see if the norm of the difference really is going as $\mathcal{O}(h^2)$.

Of course, since the halo orbits have been studied by many people, perhaps the best way of checking things is to compare with the published results. Specifically, Breakwell & Brown [1979] gave results on the halo orbits for the L_2 equilibrium point, and they were found to agree with what is given here. Howell [1983] worked with other mass ratios, but her results are also in qualitative agreement.

4.2.2 Computation of the Poincaré Map and its Derivatives

On p. 34 the Poincaré map of a periodic orbit γ was defined. Here we would like to discuss the practical numerical computation of such a map. Following Simó [1989], we will show how to compute the Poincaré map Θ and its derivatives. The calculations are given for the vector field X , but exactly the same discussion works for X_H .

In this discussion, we are working in \mathbb{R}^{2n} and there is a periodic orbit γ . For greater generality,⁴ we consider two Poincaré sections S_1 and S_2 , and a smooth map $P : S_1 \rightarrow S_2$ where $P(x) = \phi(T(x), x)$. ϕ is the flow map of the vector field, T is a positive, real valued function giving the time for the flow through the point $x \in S_1$ to first reach S_2 . This is quite similar to the Poincaré map, indeed the map P would reduce to it if $S_1 = S_2$. We specify the codimension 1 submanifolds S_1 and S_2 as $g_1^{-1}(0)$ and $g_2^{-1}(0)$, respectively, for two smooth functions $g_i : \mathbb{R}^{2n} \rightarrow \mathbb{R}$.

$$F(x, \tau - \delta(x)) = \phi(T(x), x)$$

We fix two distinct points $x_0, y_0 \in \gamma(\mathbb{R})$, and we locate S_1 at x_0 and S_2 at y_0 ; an easy way to make sure these Poincaré sections are transverse to the flow is to explicitly define them as

$$\begin{aligned} g_1(x) &= (x - x_0)^T X(x_0) \\ g_2(y) &= (y - y_0)^T X(y_0) \end{aligned}$$

⁴The reason is to set up a framework that would enable the discussion to be easily generalized to the case of multiple, or parallel, shooting.

The map P is differentiated to obtain

$$\begin{aligned} DP_x &= D_1\phi_{(T(x),x)} \cdot DT_x + D_2\phi_{(T(x),x)} \\ &= X(x) \cdot DT_x + D_2\phi_{(T(x),x)} \end{aligned}$$

Now an expression is needed for DT_x . To do this we differentiate the relation $g_2(P(x)) = 0$, using the above expression for DP_x , and find:

$$\begin{aligned} D(g_2)_y[X(x) \cdot DT_x + D_2\phi_{(T(x),x)}] &= 0 \\ DT_x &= \frac{-1}{D(g_2)_y \cdot X(x)} D(g_2)_y D_2\phi_{(T(x),x)} \end{aligned}$$

Upon substituting this expression back into the expression for DP_x we get

$$DP_x = X(x) \cdot \frac{-1}{D(g_2)_y \cdot X(x)} \cdot D(g_2)_y D_2\phi_{(T(x),x)} + D_2\phi_{(T(x),x)} \quad (4.5)$$

For our primary task of computing the main family of orbits, we abandon the more general framework and choose $S_1 = \{(x_1, x_2, x_3, x_4, x_5, x_6) \mid x_2 = 0\} = S_2$, and the map P becomes Θ . Then $g_1(x) = x_2 = g_2$.

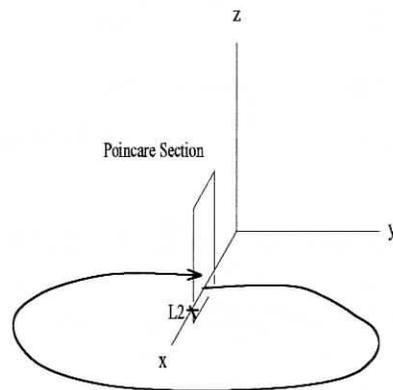


Figure 4.1: Poincaré map in configuration space

Remember, only the configuration space (x, y, z) is being shown, but the actual system, and orbits, we are considering is in \mathbb{R}^6 . This can cause some confusion.

4.2.3 Restricting the Domain of the Poincaré Map

Following Howell [1983], we make two different restrictions of the Poincaré map, one for using Newton's method to close an orbit, and another for computing the eigenvalues. Indeed, trying to use only one restricted Poincaré map for both tasks results in many problems (as is discussed in the following section). First, however, we will discuss how to make the restriction that results in efficient iterations to close an orbit.

The iteration will be done holding the x_1 component of the initial conditions fixed. (This works well for the planar family, but when trying to follow a branch of the halo orbits, we release x_1 and hold x_3 fixed instead). Define the map R by the upper route in the following diagram

$$\begin{array}{ccc}
 (x_0, 0, z_0, \dot{x}_0, \dot{y}_0, \dot{z}_0) & & (x_1, 0, z_1, \dot{x}_1, \dot{y}_1, \dot{z}_1) \\
 \mathbb{R}^6 & \xrightarrow{\phi(T(\cdot), \cdot)} & \mathbb{R}^6 \\
 \uparrow i_{x_0} & & \downarrow \pi \\
 \mathbb{R}^4 & \xrightarrow{R} & \mathbb{R}^4 \\
 \mathbf{w} = (z_0, \dot{x}_0, \dot{y}_0, \dot{z}_0) & & (z_1, \dot{x}_1, \dot{y}_1, \dot{z}_1)
 \end{array}$$

Now we define $G(w) := R(w) - w$. Then the Newton iterates are

$$w_{n+1} = w_n - DG_{w_n}^{-1}(G(w_n)) \quad (4.6)$$

The only thing that remains is to explicitly compute DG_w (we will avoid computing inverses, and instead use Gaussian elimination to solve the equivalent system $DG_{w_n}(\Delta w_n) = -G(w_n)$), where $w_{n+1} = w_n + \Delta w_n$. It is apparent that $DG_w = DR_w - I$ and from the diagram,

$$DR_w = D\pi \circ D\Theta \circ D(i_{x_0})_w = \pi \circ D\Theta_{i(w)} \circ i_{x_0}$$

Eq. (4.5) is used to get an expression for $D\Theta$, using our specific choice $g_2(x) = x_2$. What is done in the program is to first find the time T when the trajectory first returns to the Poincaré section. Then the variational equations are numerically integrated to time T , and a 6x6 matrix for $D(\phi_T)_x = D_2\phi_{(T(x),x)}$ is obtained. If the entries of this matrix are denoted a_{ij} and the components of the vector field are X^j , the end result is that

$$DR_w = \begin{bmatrix} a_{33} & a_{34} & a_{35} & a_{36} \\ a_{43} & a_{44} & a_{45} & a_{46} \\ a_{53} & a_{54} & a_{55} & a_{56} \\ a_{63} & a_{64} & a_{65} & a_{66} \end{bmatrix} - \begin{bmatrix} \frac{X^3}{X^2}a_{23} & \frac{X^3}{X^2}a_{24} & \frac{X^3}{X^2}a_{25} & \frac{X^3}{X^2}a_{26} \\ \frac{X^4}{X^2}a_{23} & \frac{X^4}{X^2}a_{24} & \frac{X^4}{X^2}a_{25} & \frac{X^4}{X^2}a_{26} \\ \frac{X^5}{X^2}a_{23} & \frac{X^5}{X^2}a_{24} & \frac{X^5}{X^2}a_{25} & \frac{X^5}{X^2}a_{26} \\ \frac{X^6}{X^2}a_{23} & \frac{X^6}{X^2}a_{24} & \frac{X^6}{X^2}a_{25} & \frac{X^6}{X^2}a_{26} \end{bmatrix}$$

Now we discuss how to compute the restriction of the Poincaré map to a submanifold where the Jacobi integral is constant. First, the function f is defined as

$$\begin{aligned} \dot{y} &= \pm \left[C - x^2 - y^2 + \frac{2(1-\mu)}{r_1} - \frac{2\mu}{r_2} + \dot{x}^2 + \dot{z}^2 \right]^{\frac{1}{2}} \\ &= +f(x, z, \dot{x}, \dot{z}) \end{aligned}$$

where the positive root is chosen, due to the direction of our trajectory (see Fig. (4.1)). Θ_e is defined by the upper route in the following diagram, observing that now $w = (x_0, z_0, \dot{x}_0, \dot{z}_0)$.

$$\begin{array}{ccc} (x_0, 0, z_0, \dot{x}_0, f(x_0, 0, z_0, \dot{x}_0, \dot{z}_0), \dot{z}_0) & & (x_1, 0, z_1, \dot{x}_1, \dot{y}_1, \dot{z}_1) \\ \mathbb{R}^6 & \xrightarrow{\phi(T(\cdot), \cdot)} & \mathbb{R}^6 \\ F \uparrow & & \downarrow \pi \\ \mathbb{R}^4 & \xrightarrow{\quad} & \mathbb{R}^4 \\ \mathbf{w} = (x_0, z_0, \dot{x}_0, \dot{z}_0) & & (x_1, z_1, \dot{x}_1, \dot{z}_1) \\ & \uparrow \Theta_e & \\ & \text{"Qe"} & \end{array}$$

We then compute $D(\Theta_e)_w$ in a very similar manner to the way DR_w was computed.

4.2.4 Newton's Method Fails When Restricted to Energy Submanifold

The natural thing to do, from a theoretical point of view, would be to use the map $D(\Theta_e)_w$ for the Newton iterations also. We could then parameterize the family of orbits by the value of the Jacobian constant, and the algorithm would be to fix such a value and use the Newton iterations to converge to a closed orbit on the constant energy surface. One would think that this simple idea would work in the numerical computations, but surprisingly, our experience shows that it does not. Even when the exact initial conditions for a closed orbit are used, this algorithm diverges away from the closed orbit.

No paper seems to give a discussion of the numerical implementation this method for the R3BP or why it should fail. The author believes that most probably he has made some type of an error, but he cannot find one. This single problem resulted in huge delays in the completion of this work. One possible explanation is the high condition number of the $D(\Theta_e)_w$ matrix, on the order of 10^5 . However, the successful other version of the Newton algorithm could handle condition numbers of DG_w as high as 10^7 . We may take some comfort, however, in the fact that Howell [1983] does not use the Newton method restricted to an energy surface either.

A more serious annoyance is the fact that the failure of this method prevents the implementation of a parallel shooting algorithm. This is because we would need to reduce the degrees of freedom in each of the multiple Poincaré

sections, and this can no longer be done by simply fixing one of the co-ordinates.

4.3 Numerical Continuation of the 1-parameter Family

After successfully refining initial conditions for a closed orbit by any algorithm, the next step is to follow along the family to larger and larger orbits. We are parameterizing the family by the x co-ordinate of the point where the orbit cuts the Poincaré section Σ , specifically on the interval between the Moon and L_2 ⁵. This is the same co-ordinate we are holding fixed during the iterations. We started very close to L_2 , using the linearized equations. There is a single vector of initial conditions in the velocity phase space that will produce periodic orbits, see eq. (4.1). We can easily find two distinct points along this vector by multiplying it by the scaling parameter δ . Then we plug each of these values into the nonlinear system and numerically refine the initial conditions until the orbits close to within a prespecified tolerance ($\epsilon = 5 \cdot 10^{-9}$). Two distinct orbits are then obtained, each with a different x component in Σ . Here are the actual co-ordinates of the first closed orbit, found from the linearized equations ($\delta = 10^{-3}$):

$$\begin{bmatrix} x_0 \\ y_2 \\ z_0 \\ \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{bmatrix} = \begin{bmatrix} 0.1155347229309 \cdot 10^1 \\ 0 \\ 0 \\ 0.3009659746387 \cdot 10^{-9} \\ 0.1816599164837 \cdot 10^{-2} \\ 0 \end{bmatrix} \quad (4.7)$$

⁵However, after the halo bifurcation is reached, we must parameterize the out-of-plane branches of the family differently.

The units are such that the Earth - Moon distance is unity. Note that x and \dot{y} are the only components of the initial conditions not equal to zero. (Actually, \dot{x} is nonzero, but very small. y is zero because we are in Σ and $z = 0 = \dot{z}$ because these orbits are contained in the x, y plane). If we assume the relationship between x and \dot{y} is given by an equation of the form

$$x = m\dot{y} + b$$

then we can solve for m and b from our two known orbits, and the linearly extrapolate a guess for a third orbit by incrementing \dot{y} and obtaining x from the above equation. However, since the linear relationship is not exact, we recompute m and b at each step. This crude method actually works satisfactorily for the planar family, but a better method is to find the first few orbits this way, and then switch to a cubic spline extrapolation. This cuts down on the number of Newton iterations required, usually taking only 2, for both the planar and halo families.

4.4 Results on the Planar and Halo Families

As mentioned before, we computed the families of orbits by integrating both the Hamiltonian vector field X_H and the Lagrangian vector field X . Breakwell & Brown [1979] and Howell [1983] both integrated only X , and indeed that is probably the most efficient choice; while computing the planar family after the point of halo bifurcation, we found that the program integrating X_H ran into difficulty long before the one integrating X did. Our primary reason for utilizing the Hamiltonian vector field was to make it easier to connect with the theoretical discussion on symmetry and bifurcations given in Chapter

3. In the following discussion, we will always refer to X_H and the co-ordinates (x, y, z, p_x, p_y, p_z) unless explicit mention is made to the contrary.

Using the linearized vector field to make a good guess for initial conditions, and refining those conditions with our algorithm, the first closed orbit is found. It is quite small and centered around L_2 , see Fig(4.2). The initial conditions are given in eq. (4.7).

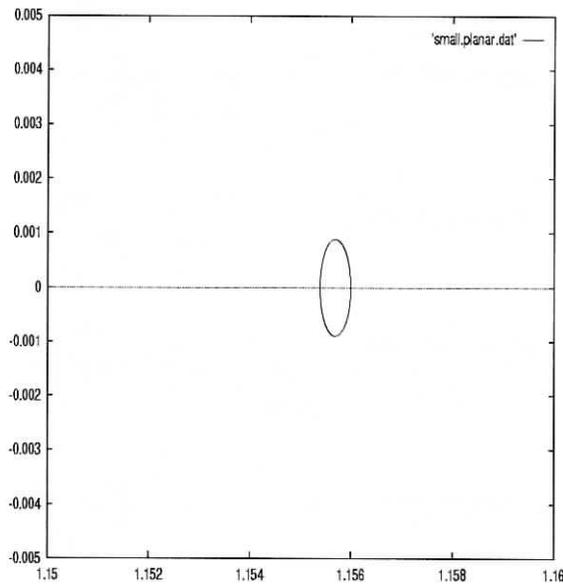


Figure 4.2: Top View of Small Planar Orbit

4.4.1 Evolution of Eigenvalues Along an Orbit

The eigenvalues give information on the linear stability of the orbit, and they make it possible to determine when the family of orbits is approaching a bifurcation point. We compute the eigenvalues using the RG subroutine from the EISPACK software library. It computes the eigenvalues and eigenvectors for a general matrix with real entries.

Given the ease and speed with which eigenvalues are computed here, an interesting phenomena can be observed. We compute all the eigenvalues of the matrix $D(\phi_t)_x$ for incremental values of $0 \leq t \leq T$, where T is the period. We want to notice the paths in the complex plane which the eigenvalues follow to get to their final destinations. See Fig. (4.3) The following computation is made using the small planar orbit that was just found.

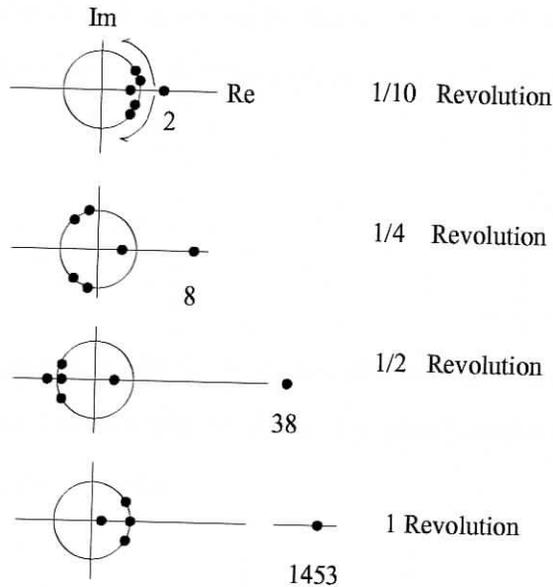


Figure 4.3: Evolution of Eigenvalues of $D(\phi_t)_x$ Around an Orbit

Since $D(\phi_{t=0})_x = Id$, all 6 eigenvalues initially are at +1. We also know that after one complete revolution along the orbit, there must be two eigenvalues at +1. It is interesting to note the pair of eigenvalues which come off the unit circle briefly at one half of a cycle. If this is merely the result of numerical errors, or if there is a deeper significance of this phenomena is unknown. After a full cycle, one eigenvalue is very close to 0. This shows that the matrix is almost singular; its condition number is $8 \cdot 10^3$. However, the eigenvalue behavior is

not really a good indicator of the condition number because as we follow the family, the eigenvalue moves away from 0, but the condition number continues to rise.

4.4.2 The Planar Family

The small planar orbit we just found and exhibited in Fig. (4.2) is symmetric. Specifically, as a set of points in the position - momentum phase space,⁶ it is invariant⁷ under the transformation we gave in Chapter 3 as

$$\mathcal{L}_{246} = \begin{bmatrix} 1 & & & & & \\ & -1 & & & & \\ & & 1 & & & \\ & & & -1 & & \\ & & & & 1 & \\ & & & & & -1 \end{bmatrix}$$

In the configuration space, the orbit makes two orthogonal crossings of the $x-z$ plane. Also since it lies entirely in the $x-y$ plane of the configuration space, it is also invariant under both

$$\mathcal{L}_{36} = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & -1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & -1 \end{bmatrix} \quad \mathcal{L}_{234} = \begin{bmatrix} 1 & & & & & \\ & -1 & & & & \\ & & -1 & & & \\ & & & -1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix}$$

So there is certainly no shortage of symmetries. These are all preserved for all members of the planar family, at least as far as we were able to compute the family numerically. See the section in Chapter 3 on symmetric periodic orbits.

⁶To make comparisons with Chapter 3 easier, this discussion is given in terms of the Hamiltonian formulation.

⁷In the initial conditions given by eq. (4.7), the 4th component is so small we assume it is only nonzero due to numerical errors.

The period of the first orbit is 3.373262718 characteristic time units (the period of the Moon's rotation is 1) and the value of the Hamiltonian is 1.5860795. The eigenvalues of the Poincaré map are

$$1453.5, \quad 0.00069, \quad 0.967 + 0.255i, \quad 0.967 - 0.255i$$

Notice the first eigenvalue. This indicates that there is quite a strong unstable direction; this does not seem to cause numerical problems, however.

As we move along the planar family, toward larger and larger orbits, the large real eigenvalue ($\lambda = 1453$) and its reciprocal partner slide slowly towards each other along the real axis. The two complex conjugate eigenvalues that started on the unit circle slide along it until they collide at +1; this is the halo bifurcation. The initial conditions of a member of the planar periodic orbit family very near this point are $x_0 = 1.12039$ and $\dot{y}_0 = 0.17604$; all other components are zero.

Continuing the planar family past the point of halo bifurcation, the complex conjugate pair moves off the unit circle at +1 and slide in opposite directions along the real axis. Near the orbit with co-ordinates (given in terms of positions and velocities) $x_0 = 1.04897$ and $\dot{y}_0 = 0.55666$, this pair of eigenvalues changes direction, and they move toward each other along the real axis. It would be interesting to know what is happening in the phase space to make this occur. The pair collides again at +1 near the orbit with co-ordinates $x_0 = 1.029$ and $\dot{y}_0 = 0.7225$. Perhaps other families of orbits branch off here; this was not investigated. The pair separates again along the unit circle. Eventually, numerical difficulties forced us to stop following this family, as the orbits got too close to the moon; it would seem that regularization is necessary, as

was employed in Howell [1983]. The initial conditions for the largest orbit we were able to compute (see Fig (4.4)) are given below (in terms of position - velocity co-ordinates, since the integrator of the Hamiltonian vector field could not even get this far)

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{bmatrix} = \begin{bmatrix} 1.01057563 \\ 0 \\ 0 \\ -0.322 \cdot 10^{-9} \\ 1.02453806 \\ 0 \end{bmatrix}$$

and the eigenvalues were

$$146.8, \quad 0.00681, \quad -0.0267 + 0.999i, \quad -0.0267 - 0.999i$$

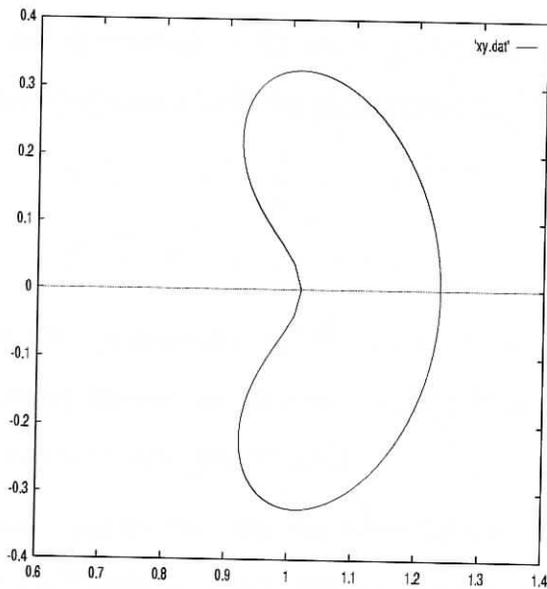


Figure 4.4: Large Planar Orbit (Moon at $x = 0.988$)

4.4.3 Stability Regions of R. Broucke

In 1969 Professor Broucke discovered an interesting way of depicting the information contained in the eigenvalues of a 4×4 real symplectic matrix M . It avoids the need for numerical computation of the eigenvalues; if the eigenvalues are desired, they can be found by solving 3 second order polynomial equations.

By exploiting certain well known properties of symplectic matrices, the characteristic equation can be written as

$$\lambda^4 + a\lambda^3 + b\lambda^2 + a\lambda + 1 = 0$$

The idea is to work directly with the coefficients a and b , called the *stability coefficients*, rather than the eigenvalues. The coefficients a and b are always real. If we denote the eigenvalues of M as $\lambda_1, \frac{1}{\lambda_1}, \lambda_2, \frac{1}{\lambda_2}$, then the relationship between the stability coefficients and the eigenvalues is

$$\begin{aligned} a &= -\left[\lambda_1 + \frac{1}{\lambda_1} + \lambda_2 + \frac{1}{\lambda_2}\right] \\ b &= 2 + \left[\lambda_1 + \frac{1}{\lambda_1}\right] \cdot \left[\lambda_2 + \frac{1}{\lambda_2}\right] \end{aligned}$$

In the present paper, we compute the stability coefficients *from* the eigenvalues, since professional quality eigenvalue routines of EISPACK were available, and sometimes the eigenvectors were also wanted.

If the stability coefficients are available, instead of drawing a picture of the eigenvalues for a family of orbits, we can draw a picture of the a and b values. The a, b plane is divided into 7 regions by curves that delimit regions corresponding to qualitatively different eigenvalue configurations. We get the following picture, taken from Howard & MacKay [1987].

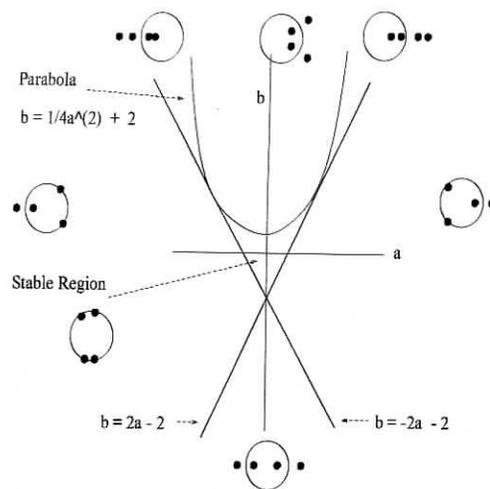


Figure 4.5: 7 stability regions and eigenvalue configurations

In Fig (4.6) the curves of values of a and b that are obtained from the planar and halo families are superimposed. Note that the diagonal straight line is the boundary line $b = -2a - 2$. The curve that starts near the upper left hand corner, and is above this line for most of its duration, is the trace of the planar family. The upper point where it crosses the boundary line corresponds to the halo bifurcation, and the lower crossing point corresponds to the other collision of eigenvalues at $+1$. The lower curve, which is born at the halo bifurcation and which goes very close to the origin, corresponds to the halo family of orbits. Note the curious wobbly behavior of this curve near the origin. In Fig (4.7) a close up of this region is shown, where the calculated data points are not connected by line segments. The discontinuous behavior is troubling, and most likely results from instabilities in the eigenvalue calculation routines. This point calls for further study, perhaps by working directly with the stability coefficients and not obtaining them from the eigenvalues.

The halo curve enters the stable region briefly and then leaves it again as the family of orbits approach a collision with the lunar surface.

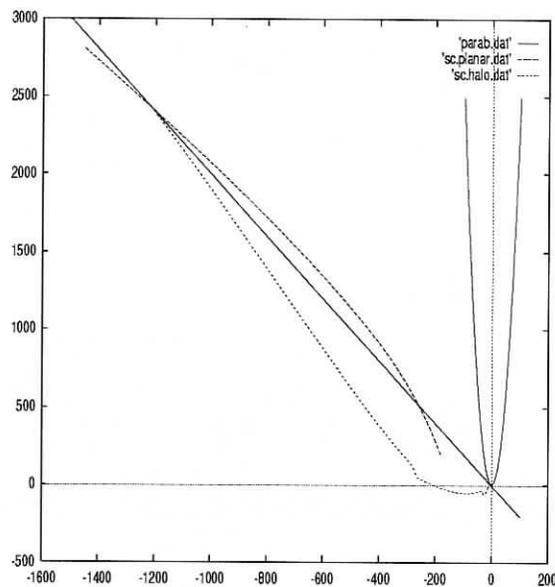


Figure 4.6: Trace of Planar and Halo Families

For more information on stability regions and coefficients, the reader may refer to Broucke [1969], Howard & MacKay [1987], Hadjimetriou [1975], Contopoulos & Barbanis [1994].

4.4.4 Analysis of Halo Bifurcation

An important difference between the numerical computations done here and those done in Breakwell & Brown [1979] and Howell [1983] is that the algorithm we use does not *presuppose* the symmetry of all members of the families of orbits. Specifically, we integrate the trajectories for a full period, rather than the half period that the \mathcal{L}_{246} symmetry would allow us to use. The reason for this is that the author did not know, *a priori*, when and how symme-

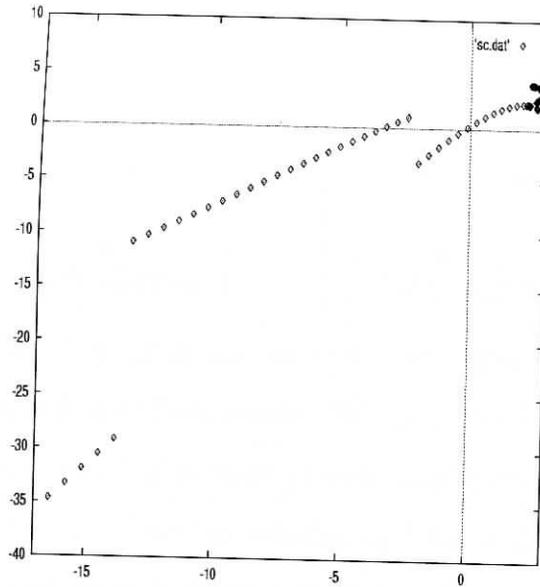


Figure 4.7: Close Up of Halo Family Near Origin

try breaking might occur, therefore we took a more conservative approach. In hindsight, it should be noted however, that we did not find anything different than was contained in the two papers mentioned above.

The bifurcation was found to occur very near the orbit with initial conditions

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{bmatrix} = \begin{bmatrix} 1.120385629610 \\ 0 \\ 0 \\ 0 \\ 0.1760447949491 \\ 0 \end{bmatrix}$$

The eigenvectors of the *monodromy matrix* $D(\phi_T)_x$, associated to the two eigen-

values that just collided at +1 were

$$\begin{bmatrix} 0 \\ 0 \\ -15.42867499 \\ 0 \\ 0 \\ -0.264840664 \end{bmatrix} \quad \begin{bmatrix} 0 \\ 0 \\ 30.206831440 \\ 0 \\ 0 \\ -0.518515213 \end{bmatrix}$$

These give us the directions to move our initial conditions in the phase space to move onto the branches of halo orbits. We chose to follow the first eigenvector. First, however, let us give some justification as to why following these eigenvectors should lead us along the halo family.⁸ Let A denote the map DP_x . We know $P(x) = x$; we want to find a vector h such that $P(x+h) = x+h$. If h is an eigenvector of A with eigenvalue +1, $Ah = h$. By definition of the differential map

$$\lim_{\|h\| \rightarrow 0} \left[\frac{\|P(x+h) - P(x) - Ah\|}{\|h\|} \right] = 0 \quad (4.8)$$

So, given $\epsilon > 0$, there exists $\delta > 0$ such that $\|P(x+h) - P(x) - Ah\| < \epsilon\|h\|$. Now since $Ah = h$, $\|P(x+h) - (x+h)\| < \epsilon\|h\|$. This says that if $\|h\|$ is small, $(x+h)$ is close to being another fixed point.

Symmetry Breaking Based on the results of Rimmer, given in Chapter 3, we would expect the planar family of orbits to lose its symmetry on the halo branches. The situation is somewhat confusing, however, because there are three symmetries occurring simultaneously! The symmetry that seems to be most often talked about in the literature is \mathcal{L}_{246} ; this is what is referred to by

⁸If we stay with the same continuation method we were using, we will move easily through this bifurcation point, and stay with the planar family

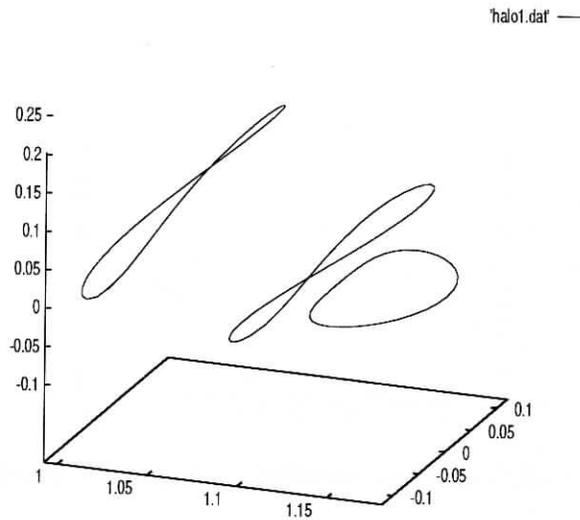


Figure 4.8: Planar Bifurcation Orbit and Halo Orbits

Howell [1983] when she states that the halo orbits are symmetric. However, this is the *only* symmetry inherited by the halo family; both \mathcal{L}_{36} and \mathcal{L}_{234} are broken. Therefore, we have a situation that seems to fit into the categories discovered by Rimmer, but there is more to it. Multiple symmetries in the 2 dimensional case are discussed somewhat in Aguiar, Malta, et. al. [1987].

4.4.5 The Halo Family

If we move onto a halo branch when the two conjugate eigenvalues collide at +1, the eigenvalues do not leave the unit circle, they pass through each other. There are two branches, but it is sufficient to follow only one of them, due to the symmetry discussed in the previous chapter.

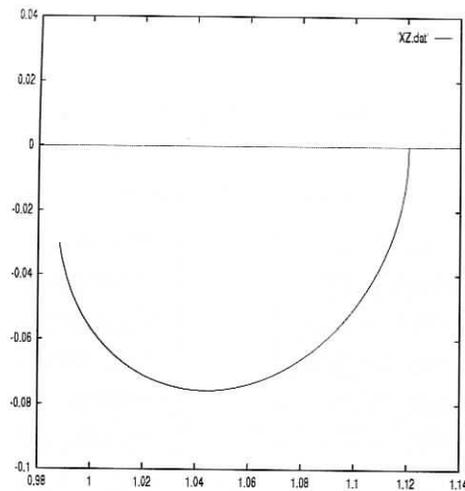


Figure 4.9: x, z Co-ordinates of Halo Orbits

It is a little tricky to numerically continue the halo family because of the behavior of the curve of initial conditions of the orbits. As previously mentioned, the only co-ordinates that are nonzero are x_0, z_0 and \dot{y}_0 . The later co-ordinate is monotonically increasing, but the turning of the x, z curve in Fig (4.9) made it necessary first to hold z_0 ⁹ fixed during the orbit refining iterations, and later on we had to switch to holding x_0 fixed. This is all very simple once Fig (4.9) is known, but quite difficult without it.

There are several further bifurcation points along the halo family which have been identified. After the collision of the pair of eigenvalues at $+1$, where the halo orbits originated, the two eigenvalues remain on the unit circle, seemingly passing through each other. They slide along toward a collision at

⁹Also referred to as x_3 in some parts of this paper.

-1, which occurs at

$$x_0 = 1.00720981028, \quad z_0 = -0.0635487960693, \quad \dot{y}_0 = 0.539728830441$$

The two eigenvectors associated with the colliding eigenvalues are

$$\begin{bmatrix} -0.093176 \\ 6.468029 \\ -0.305020 \\ -9.894446 \\ -1.393888 \\ 20.698119 \end{bmatrix} \quad \begin{bmatrix} -0.303282 \\ -21.052949 \\ -0.992817 \\ 32.20568 \\ -4.537001 \\ -67.37082 \end{bmatrix}$$

Other branches of periodic orbits may originate here, but there are several technical difficulties which must be overcome in order to find them. Note that the 2nd components are nonzero, indicating that the Poincaré section we have been using, $S = \{y = 0\}$, may no longer be sufficiently transverse to the new family. Also, since this would be a period doubling bifurcation, the orbits in the new family will have a period twice as long as the halo orbit. This presents a practical problem in that the condition number of the DG_w grows rapidly with the integration time. The period of the halo orbit is 2.763470 and the condition number is already up to $1.75 \cdot 10^4$, so if the integration time is doubled, the condition number should be about 10^8 , which will probably cause the Newton iterations to fail to converge. A natural way to overcome this problem is to use multiple shooting, but for that we need to restrict to a fixed energy surface. As has been discussed earlier, that seems to inevitably fail.

The pair of eigenvalues under discussion now pass through each other at -1 and stays on the unit circle. The real pair of eigenvalues now collides at +1 at the orbit

$$x_0 = 0.9924987045, \quad z_0 = -0.04500163013, \quad \dot{y}_0 = 0.6867405173$$

The two eigenvectors associated with the colliding eigenvalues are found to be (probably indicating great numerical instability)

$$\begin{bmatrix} -0.000388 \\ 0.994429 \\ 0.0008357 \\ 1.1296156 \\ 0.0075741 \\ 8.6141233 \end{bmatrix} \quad \begin{bmatrix} 13621.9 \\ 34925434.0 \\ -29358.8 \\ 39684700 \\ -266086.4 \\ 302624100 \end{bmatrix}$$

The author attempted to find a branching family in the direction indicated by one of the eigenvalues, but again was unsuccessful. Here the problem is not the integration time, but a modified algorithm to refine initial conditions would seem to be necessary. The Jacobi constant also reaches a minimum value here of 3.01517757.

All eigenvalues of the halo family are now on the unit circle, so the halo orbits in the region are linearly stable. The pair that just entered at +1 slides rapidly toward -1, while the other pair moves slowly toward +1. As the first pair gets close to the vertical axis, the second pair reverses direction, and collides at -1 again, moving off the unit circle.

A note of caution is that discontinuities were observed in the curve of stability coefficients, and this causes some suspicion of the accuracy of the computation of the eigenvalues. However, the results found here seem to agree with Breakwell & Brown [1979].

The largest halo orbit that we computed had initial conditions

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{bmatrix} = \begin{bmatrix} 0.98796165 \\ 0 \\ -0.029770651 \\ 0 \\ 0.86446415 \\ 0 \end{bmatrix}$$

In our R3BP model, the Moon is a point mass, but close to this orbit the halo family would intersect the surface of the actual Moon; thus following the family further would seem to not be very useful.

Chapter 6

Appendix: Source Code

6.0.6 Flowchart

