Im folgenden Programmsegment wird zuerst $\bar{\omega}$ berechnet und in einer Zwischenvariablen obar gespeichert. Damit wird der neue Winkel $\varphi(t + \Delta t)$ berechnet, $\bar{\varphi}$ wird in einer Zwischenvariablen pquer abgelegt und zur die Berechnung von ω benutzt. Die Zeit t wird zweimal um $\frac{\Delta t}{2}$ erhöht und somit bei jedem Durchlauf um Δt.

```
   const int nx=10001;  // fieldlength
   double p[nx], dp[nx], t;

// parameter definition
   double g, l;        // g=gravity, l=length of pendulum
   double q, om, am;   // q=friction, om=driving frequency, am=amplitude
   double p0, dp0;     // p0=initial angle, dp0=initial angular velocity
   double t_max, dt;   // t_max=total time, dt=time step
   double fp, obar, pbar;

// read parameters
   cout << "Gravitation    ";  cin >> g;
   cout << "Pendellaenge   ";  cin >> l;
   cout << "Reibung        ";  cin >> q;
   cout << "Anreg.Frequenz ";  cin >> om;
   cout << "Anreg.Amplitude";  cin >> am;
   cout << "Anf.Auslenkung ";  cin >> p0;
   cout << "Anf.Geschw.    ";  cin >> dp0;
   cout << "Gesamtzeit     ";  cin >> t_max;
   cout << "Zeitintervall  ";  cin >> dt;
   double c=g/l;
   double dt2=dt/2.;

// make sure that the number of intervals is smaller than nx
   if( t_max/dt >= nx) {
      cout << "ERROR: dt is too small, set to: " << t_max/nx << '\n';
      return -1;
   }

// setup initial conditions
   p[0]=p0;
   dp[0]=dp0;
// loop from t=0 to t=t_max
   t=0.0;
   for( int i=1; i<t_max/dt; i++ ) {
        fp=-c*sin(p[i-1])-q*dp[i-1]+am*sin(om*t);
        obar=dp[i-1]+dt2*fp;
        p[i]=p[i-1]+dt*obar;
```

```
      t=t+dt2;
      pbar=p[i-1]+dt2*dp[i-1];
      fp=-c*sin(pbar)-q*obar+am*sin(om*t);
      dp[i]=dp[i-1]+dt*fp;

      t=t+dt2;
   }

// output to file
   ofstream outfile("ruku.dat");
   t=0.0;
   for( int i=0; i<=t_max/dt; i++ ) {
      outfile << t << ' ' << p[i+1] << ' ' << dp[i+1] << '\n';
      t=t+dt;
   }
   outfile.close();
}
```

3.2.4 Chaos

Um die Ergebnisse des Progamms besser untersuchen zu können, legen wir folgende Werte fest: $g = l = 9.81$, $\Omega = 2/3$, $q = 0.5$. Die einzigen jetzt noch frei wählbaren Variablen sind: F_0 und φ_0. In Abbildung 3.7 sehen wir den zeitlichen Verlauf des Auslenkwinkels φ für verschiedene Zeitschritte Δt und verschiedene Amplituden F_0 zwischen 0 und 1.3.

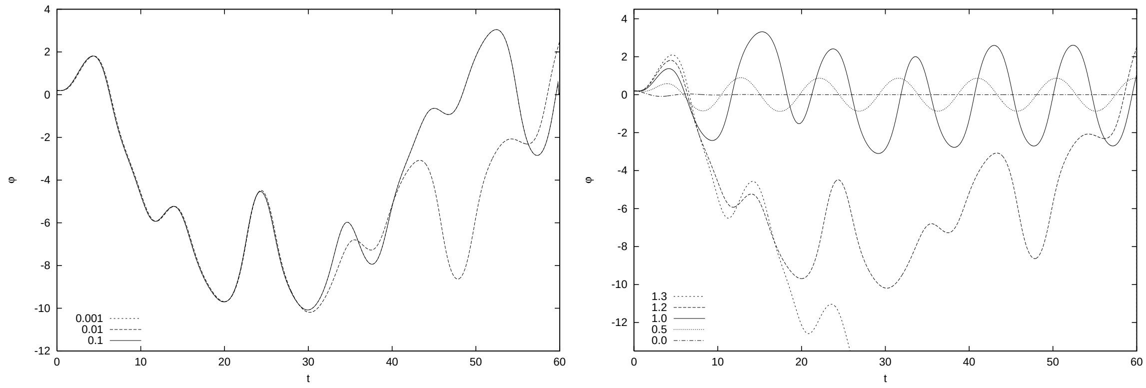
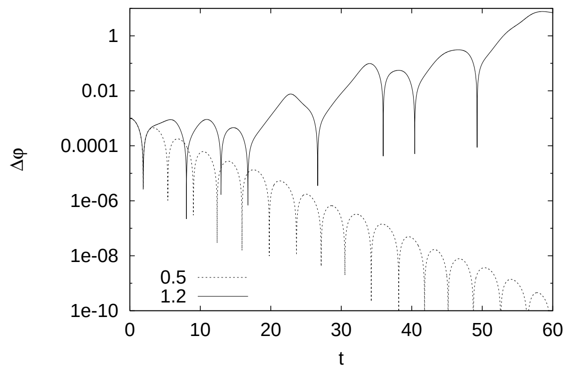


Abbildung 3.7: Links: Pendelausschlag φ als Funktion der Zeit t für verschiedene Δt. Rechts: φ als Funktion von t für verschiedene F_0.

Ist die Genauigkeit $\Delta t \leq 0.01$ so fallen die Kurven aufeinander, bei größerem Δt stellt man starke Abweichungen fest. Ohne externe Anregung $F_0 = 0$, kommt das Pendel aufgrund der Reibungskraft q schon nach kurzer Zeit zum Stillstand. Für einen kleinen Wert, zum Beispiel $F_0 = 0.5$, entsteht das gleichmäßige Schwingungsbild einer periodischen

Bewegung. Die Frequenz entspricht jedoch nicht der Eigenfrequenz des Pendels, sondern entspricht ungefähr der Frequenz der antreibenden Kraft. Wird die Antreibkraft weiter erhöht, dann bewegt sich das Pendel nicht mehr gleichmäßig. Die Ausschläge lassen sich nicht mehr vorraussagen, das Verhalten des Pendels wird *chaotisch*. Es ist bemerkenswert, daß trotz des durch die Rechnung gegebenen Determinismus das Verhalten des Pendels chaotisch ist.

Als nächstes halten wir den Wert von F_0 fest und verändern den anfänglichen Auslenkwinkel φ_0. Im ersten Fall sei der Anfangsauslenkwinkel $\varphi_{01} = 0.2$, im zweiten Fall $\varphi_{02} = 0.201$. Der Unterschied beträgt also nur 0.5 Prozent. Die Differenz ("damage spreading" genannt) dieser beiden Auslenkwinkel $\Delta\varphi(t) = \varphi_2(t) - \varphi_1(t)$ wird nun im zeitlichen Verlauf untersucht. Die Grafiken der Abbildung 3.8 sind halblogarithmische Darstellungen, d.h. die x-Achse hat einen normalen, linearen Maßstab, während die y-Achse logarithmisch ist. Der Abstand zwischen den Auslenkungen wurde einmal mit der Euler-Cromer Methode (links) und dann mit dem Runge-Kutta Verfahren (rechts) berechnet. Die jeweils obere Kurve zeigt den chaotischen Fall: Der Fehler vergrößert sich, im Mittel auf einer steigenden Geraden. Eine Linie mit positiver Steigung bedeutet in der halblogarithmischen Darstellung, daß die Funktion exponentiell wächst, der Abstand läßt sich also durch $\Delta\varphi \approx e^{\lambda t}$, mit $\lambda > 0$ nähern. Im periodischen Fall (untere) verringert sich der Abstand in der Darstellung linear, also wieder exponentiell, aber diesmal mit negativer Steigung $\lambda < 0$. λ wird als der Lyapunovexponent bezeichnet. Bei $\lambda = 0$ ist der Übergang vom Periodischen ins Chaotische mit $\lambda > 0$.



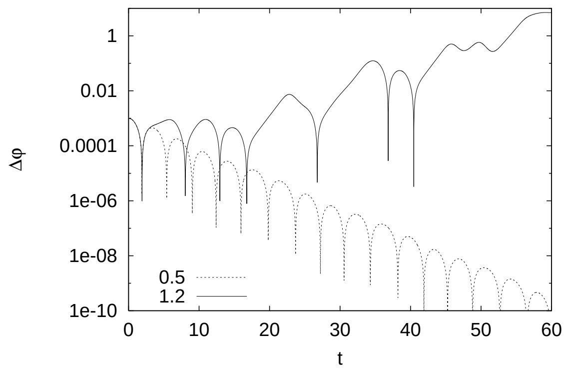


Abbildung 3.8: Damage Spreading: Der Abstand zwischen den Anfangsauslenkungen $\varphi_{01} = 0.2$ und $\varphi_{01} = 0.201$ wurde links mit Euler-Cromer, rechts mit Runge-Kutta berechnet.

Der Poincaréschnitt

Eine andere Darstellung der Pendelbewegung ist durch die Abbildung in einem sog. *Phasendiagramm* möglich. Ein Phasendiagramm ist ein Geschwindigkeit-Ort-Diagramm, in das die Winkelgeschwindigkeit ω über den Auslenkwinkel φ aufgetragen wird. Für den harmonischen Oszillator entsteht dann als Bild eine Ellipse. Diese geschlossene Kurve zeigt die Energieerhaltung beim harmonischen Oszillator. Für das allgemeine Pendel entsteht im periodischen Fall nach der Einschwingzeit ebenfalls eine Ellipse (linke Grafik, Abbildung 3.9). Im chaotischen Fall weist das Phasendiagramm eine endlose Anzahl unterschiedlicher Linien auf, die irgendwann (fast) das ganze Diagramm überdeckt haben (rechte Grafik).

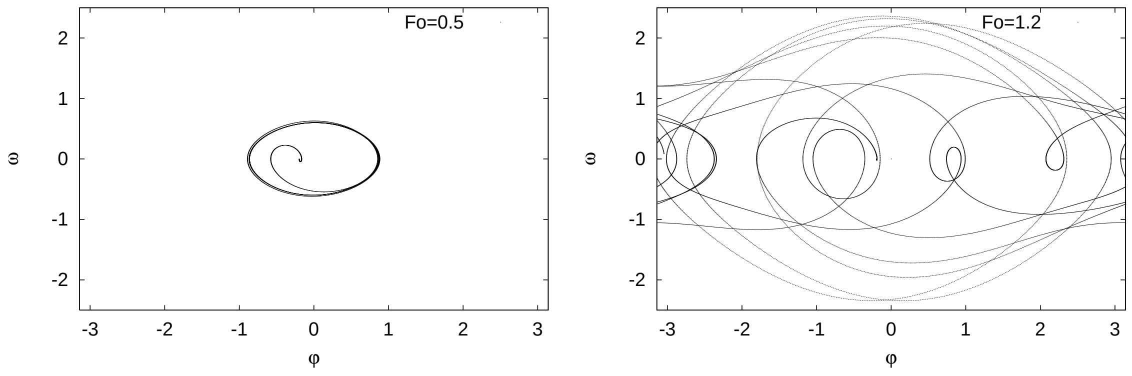


Abbildung 3.9: Phasendiagramme des allgemeinen Pendels, links für $F_0 = 0.5$, rechts für $F_0 = 1.2$.

Die Untersuchung des Bildes im chaotischen Fall ist sehr schwierig. Deswegen wird hier meist eine stroboskopische Analyse durchgeführt: Es werden nur Punkte in das Diagramm gezeichnet, die zu der gleichen Phase der angelegten periodischen Kraft vorkommen, beispielsweise für $\Omega t = 0,\ 2\pi,\ 4\pi$ etc.. Diese stroboskopischen "Schnitte" werden *Poincaréschnitte* genannt. Für den harmonischen Oszillator entsteht in dieser Darstellungsart im Diagramm nur ein einziger Punkt. Das entstehende Bild wird *Attraktor* genannt. Dieser erste Fall ist der punktförmige Attraktor. Ist die Bewegung chaotisch, dann entsteht eine Menge von Punkten. Dieses Bild heißt dann seltsamer oder fraktaler Attraktor. Für unser Pendel ist dies in der Abbildung 3.10 links zu sehen. Als drittes gibt es noch die geschlossenen Attraktoren: ein geschlossener Linienzug im Poincaréschnitt. Für ein Doppelpendel (ein Pendel, das an der Masse eines anderen Pendels aufgehängt ist) ensteht ein Diagramm, bei dem gleichzeitig zwei verschiedene Attraktoren, d.h. zwei verschiedene Fälle im gleichen Phasenraum, gezeigt werden können, ein geschlossener und ein fraktaler Attraktor.

Übergang periodisch – chaotisch

Betrachtet man das Pendel für noch größere Antreibkräfte, entsteht wieder periodisches Verhalten. Abbildung 3.11 zeigt das Pendelverhalten für die Antreibkräfte $F_0 = 1.35$, $F_0 = 1.44$ und $F_0 = 1.465$. Es fällt auf, daß die Perioden für die letzten beiden Werte doppelt, bzw. viermal so groß sind. Die Periode wird also jeweils verdoppelt. Dieses Verhalten wird Bifurkation genannt.

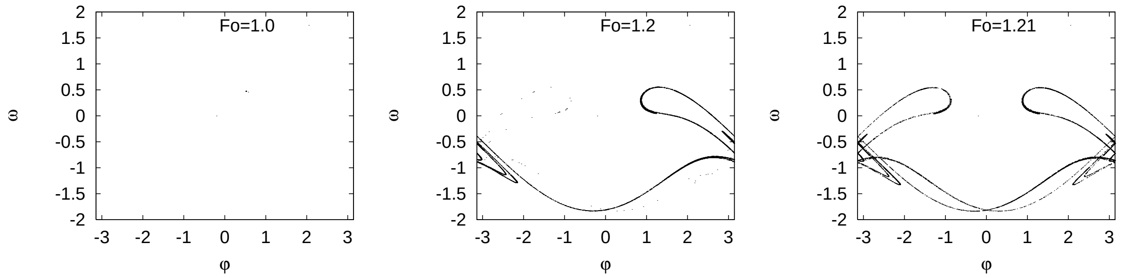


Abbildung 3.10: Poincaréschnitte für das allg. Pendel

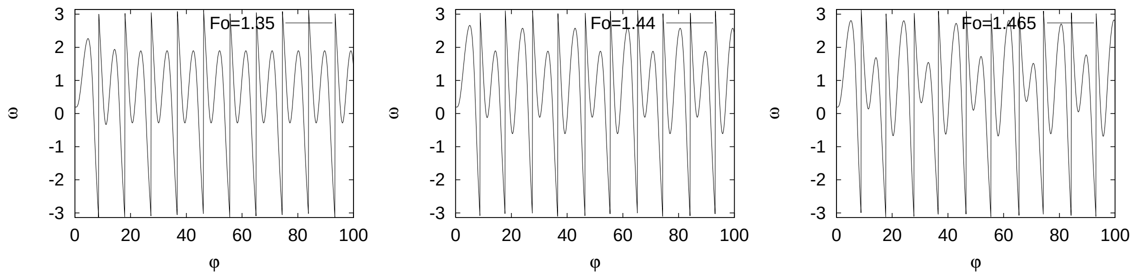


Abbildung 3.11: Unterschiedliche Perioden bei verschieden starken Antreibkräften

Das Bifurkationsdiagramm aus Abbildung 3.12 zeigt die Punkte des Poincaréschnittes in Abhängigkeit der Antreibfrequenz F_0. Die Bifurkationen sind in diesem Fall an den Stellen $F_0 \approx 1.43$ und $F_0 \approx 1.46$. Ab dem Wert $F_0 \approx 1.47$ beginnt das Chaos. Dieser Übergang zum Chaos wird auch Periodenverdopplungs– oder Kaskadenszenario genannt.

Die Längen der Perioden sind im Poincaréschnitt durch die Anzahl der Punkte gegeben. Eine zweifache Periode entspricht so z.B. einem Attraktor aus zwei Punkten. Sei jetzt f_n der Wert, an dem eine Bifurkation zu 2^n besteht, d.h. das Diagramm sich in 2^n Linien aufspaltet. Dann ist die Funktion

$$\delta_n = \frac{f_{n-1} - f_n}{f_n - f_{n+1}} \tag{3.30}$$

das Verhältnis der Abstände zwischen zwei aufeinanderfolgenden Bifurkationen. δ_n geht für große n gegen den Wert 4.669. Dieser Wert ist universell, d.h. für alle Zweier-Kaskadenszenarios derselbe. Man nennt ihn Feigenbaumkonstante.

3.3 Himmelsmechanik

Im Folgenden betrachten wir die Berechnung der Planetenbewegung, zunächst einmal der Erde, um die Sonne. Entscheidend hierbei ist die Gravitationskraft zwischen der Masse

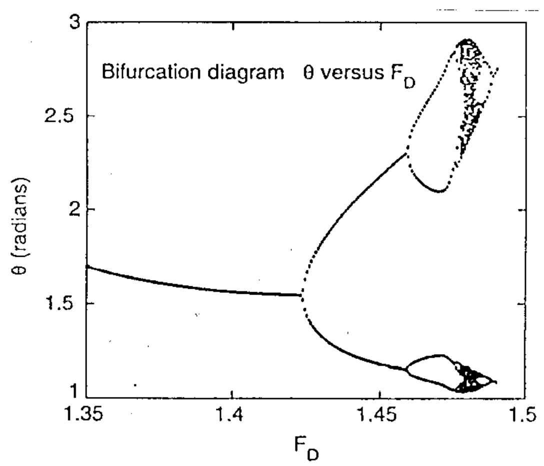


Abbildung 3.12: Bifurkationsdiagramm des Pendels

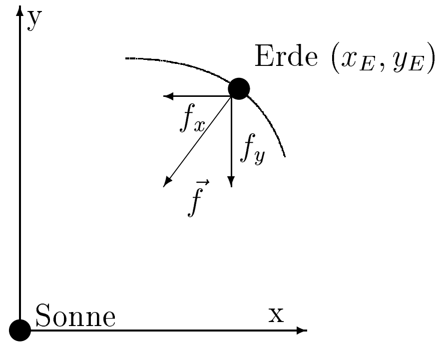


Abbildung 3.13: System Erde–Sonne

der Sonne (m_S) und der Masse der Erde (m_E). Die Formel für diese Kraft lautet :

$$\vec{f} = -\frac{Gm_Sm_E}{r^2} * \frac{\vec{r}}{r} \tag{3.31}$$

Der Faktor G ist die Gravitationskonstante und der Vektor $\vec{r} = (x,\, y)$ ist der Abstandsvektor zwischen den Massenschwerpunkten von Sonne und Erde. Zur Vereinfachung betrachten wir die Bewegung nur im Zweidimensionalen. Für die Berechnung benutzen wir die Formel in der Komponentenschreibweise für die x-Richtung f_x und in die y-Richtung f_y mit

$$f_x = -\frac{Gm_Sm_Ex}{r^3}\;,\; f_y = -\frac{Gm_Sm_Ey}{r^3} \text{ und } r = \sqrt{x^2+y^2} \tag{3.32}$$

Um geschicktere und angenehmere Zahlen zu bekommen, verwenden wir hier nicht das MKS-System (Meter–Kilo–Sekunde), sondern die Astronomische Einheit (1 AE= $1.5 * 10^{11}$ m≈ Abstand Erde – Sonne) für Längen und das Jahr (1 a) für die Zeit. Die Bahngeschwindigkeit bei der Kreisbewegung ist somit $v_0 = 2\pi * \frac{1\text{AE}}{1\text{a}}$. Da die Gravitationskraft betragsmäßig gleich der Zentrifugalkraft ist, haben wir

$$f = \frac{m_E v_0^2}{r} = \frac{G m_S m_E}{r^2} \Rightarrow G m_S = v_0^2 r = 4\pi^2 \frac{\mathrm{AE}^3}{\mathrm{a}^2} \tag{3.33}$$

Für die numerische Berechnung verwenden wir wieder die Newtonsche Bewegungsgleichung

$$\ddot{x} = \frac{f_x}{m_E} = -\frac{4\pi^2 x}{r^3} \text{ und } \ddot{y} = \frac{f_y}{m_E} = -\frac{4\pi^2 y}{r^3} \tag{3.34}$$

Wir benutzen in diesem Fall wieder die Verlet-Methode für unser Programm und lösen für die numerische Rechnung die beiden Gleichungen

$$x(t+\Delta t) = -x(t-\Delta t) + 2x(t) - 4\pi^2 x(t) r_3(t) \Delta t^2 \tag{3.35}$$

$$y(t+\Delta t) = -y(t-\Delta t) + 2y(t) - 4\pi^2 y(t) r_3(t) \Delta t^2 \tag{3.36}$$

Dabei ist die Abkürzung $r_3(t) = \left(\sqrt{x(t)^2 + y(t)^2}\right)^{-3}$ verwendet worden.

```
int main()
{
   double p0x, p1x, p2x, dp0x, dp1x, dp2x, t;
   double p0y, p1y, p2y, dp0y, dp1y, dp2y, r3;
   double alpha, t_max, dt;

// read parameters
   cout << "Anf.x  ";  cin >> p1x;
   cout << "Anf.y  ";  cin >> p1y;
   cout << "Anf.vx ";  cin >> dp1x;
   cout << "Anf.vy ";  cin >> dp1y;
   cout << "Alpha  ";  cin >> alpha;
   cout << "Gesamtzeit     ";  cin >> t_max;
   cout << "Zeitintervall  ";  cin >> dt;

   const double beta=(alpha)/2.;
   const double dt2=dt*dt;
   const double pi24=4.*M_PI*M_PI;

// setup initial conditions
   p0x=p1x-dt*dp1x;
   p0y=p1y-dt*dp1y;
   dp0x=dp1x;
   dp0y=dp1y;

// loop from t=0 to t=t_max
```

```
    t=0.0;
    ofstream outfile("orbit.dat");
    for( int i=0; i<t_max/dt; i++ ) {
        r3=1./pow(p1x*p1x+p1y*p1y,beta);
        p2x=2.*p1x-p0x-pi24*dt2*r3*p1x;  // Verlet algorithm
        p2y=2.*p1y-p0y-pi24*dt2*r3*p1y;
        outfile << t << ' ' << p1x << ' ' << p1y << '\n';
        t=t+dt;
        p0x=p1x;                          // change new->old
        p0y=p1y;
        p1x=p2x;
        p1y=p2y;
    }
    outfile.close();
}
```

Für ideale Anfangsbedingungen, das wäre $x_0 = 1$, $y_0 = 0$, $vx_0 = 0$ und $vy_0 = 2\pi$, erhält man für die Erdbewegung eine Kreisbahn, siehe Abbildung 3.14 (links). Verändert man die Anfangsgeschwindigkeit auf $vy_0 = 5$, so ergibt sich eine elliptische Bahn (rechts).

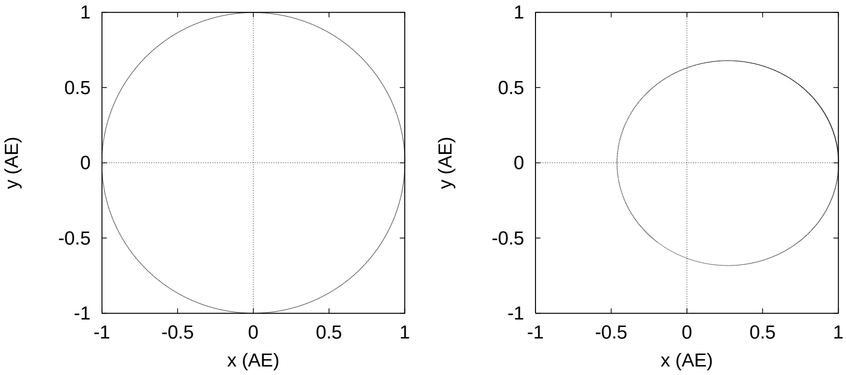


Abbildung 3.14: Bahnkurven der Erde um die Sonne für verschiedene Anfangsgeschwindigkeiten $vy_0 = 2\pi$ (links) und $vy_0 = 5$ (rechts).

Eine weitere Variationsmöglichkeit besteht in der Änderung des Gravitationsgesetzes. Anstatt die Gravitationskraft proportional zu $1/r^2$ abfallen zu lassen, kann man sie auch proportional zu $1/r^{\alpha-1}$ machen. Allgemein wird im obigen Programm `r=1/sqrt(pow(r, alpha))` verwendet, wobei `pow(r, alpha)` der Potenzierung r^α entspricht. Abbildung 3.15 zeigt vier verschiedene Beispiele mit Werten für β zwischen 3.00 und 2.01, wobei $\beta = \alpha - 1$. Die Rotation der Ellipse bei den Beispielen für $\beta \approx 2$ nennt man Präzesion.

Für $\beta = 3.00$ fällt auf, daß die Erde, nachdem sie in die Nähe der Sonne "geflogen" ist, sich mit hoher Geschwindigkeit aus dem System entfernt. Dies ist allerdings ein numerisches Problem. Für diesen Effekt ist die Wahl eines zu großen Δt verantwortlich, daß bei großen

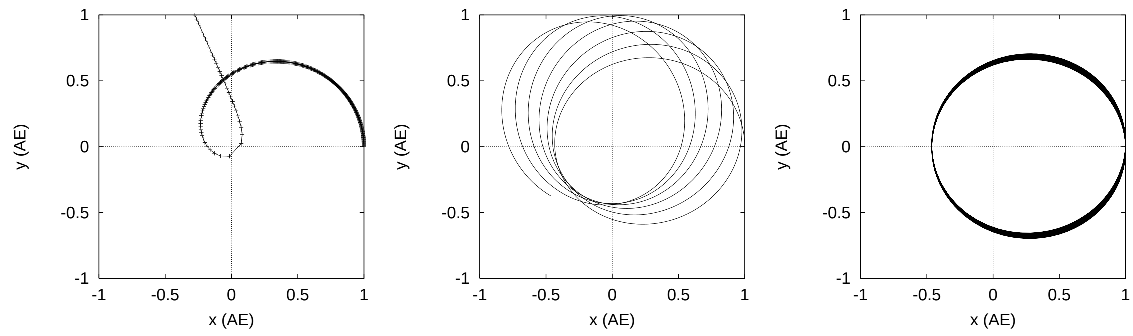


Abbildung 3.15: Bahnkurven bei versch. Gravitationsgesetzen $\beta = 3$ (links), $\beta = 2.1$ (mitte) und $\beta = 2.01$ (rechts).

Geschwindigkeitsänderungen kleiner sein sollte. Allerdings würde bei einem kleineren Δt die Simulation zu lange dauern. Man benötigt also ein automatisches Verfahren, das die Größe von Δt regelt. Eine Möglichkeit ist es, die Änderung der Energie für verschiedene Δt zu vergleichen und den Zeitschritt dementsprechend zu verändern (adaptives Verfahren). Dazu bietet sich z.B. das Runge-Kutta Verfahren an, da dort sowieso Werte zur Zeit $t + \Delta t/2$ berechnet werden. Die Energie der Erde ist

$$E_{Erde} = -\frac{Gm_S m_E}{r} + \frac{m_E}{2} v^2 \tag{3.37}$$

Die Energie ergibt sich also aus den Koordinaten x und y für $r = \sqrt{x^2 + y^2}$ direkt und für $\vec{v} = \Delta\vec{r}/\Delta t$ indirekt. Wir berechnen zuerst die Energie E_1 aus

$$x_1 = x(t + \Delta t) = f_{\Delta t}(x(t), y(t)) \tag{3.38}$$
$$y_1 = y(t + \Delta t) = g_{\Delta t}(x(t), y(t)) \tag{3.39}$$

und die Energie E_2 mit halb so großem Δt durch

$$x_2 = f_{\frac{\Delta t}{2}}(x(t + \frac{\Delta t}{2}), y(t + \frac{\Delta t}{2})) \tag{3.40}$$
$$y_2 = g_{\frac{\Delta t}{2}}(x(t + \frac{\Delta t}{2}), y(t + \frac{\Delta t}{2})) \tag{3.41}$$

Die Funktionen f und g berechnen die neuen x- bzw. y-Werte für Δt oder $\Delta t/2$. Das entscheidende Kriterium ist die Differenz

$$\Delta E = |E_2 - E_1| \tag{3.42}$$

Ist ΔE groß, so ist Δt zu klein gewählt, ist ΔE sehr klein, so kann Δt größer gewählt werden. Im Programm könnte z.B. bei $\Delta E > 10^{-5} \rightarrow \Delta t' = \Delta t/2$ und bei $\Delta E < 10^{-7} \rightarrow \Delta t' = \Delta t * 2$ gewählt werden.

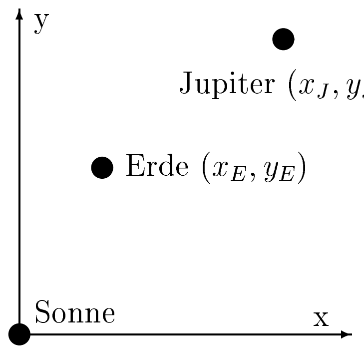


Abbildung 3.16: System Erde–Jupiter

Drei–Körper–Problem

Kommt zu den Körpern Sonne und Erde noch ein dritter (beispielsweise der Jupiter) hinzu, so ist dieses Problem nicht mehr analytisch lösbar. Die Keplerschen Gesetze zur Planetenbewegung gelten nicht mehr. Allgemein ist das beim N-Körperproblem ($N > 2$) so. Numerisch läßt sich dieses Problem immer noch lösen. Wir benötigen als erstes das Gravitationsgesetz für Erde-Jupiter:

$$f_{EJ} = -\frac{Gm_E m_J}{r_{EJ}^2} \text{ mit } r_{EJ} = \sqrt{(x_E - x_J)^2 + (y_E - y_J)^2} \tag{3.43}$$

Wieder stellen wir für die Erdbewegung (und auch für die Jupiterbewegung) eine komponentenweise Gleichung auf, die analog für y gilt.

$$\ddot{x}_E = -\frac{4\pi^2 x_E}{r_E^3} - \frac{Gm_J(x_E - x_J)}{r_{EJ}^3} \tag{3.44}$$

$$\ddot{x}_J = -\frac{4\pi^2 x_J}{r_E^3} - \frac{Gm_E(x_J - x_E)}{r_{EJ}^3} \tag{3.45}$$

Der rechte Ausdruck ist der Einfluß vom dritten Körper, also jeweils des Jupiters, bzw. der Erde. Für die Rechnung benötigen wir jetzt auch die Massen der einzelnen Körper.

$$\begin{aligned} m_E &= 6 * 10^{24}\text{kg} \\ m_S &= 2 * 10^{30}\text{kg} \\ m_J &= 2 * 10^{27}\text{kg} * M_J \end{aligned}$$

Den Faktor M_J benutzen wir später in der Simulation um auszuprobieren, wie sich unser System verhalten würde, wenn die Masse des Jupiters geändert würde. Der Radius der Jupiterbahn ist $r_J = 5.2\,\text{AE}$ und seine Bahngeschwindigkeit ist $v = 2.76\,\text{AE/a}$. Man kann nun die Konstanten

$$Gm_E = 12 * \pi^2 * 10^{-6} \tag{3.46}$$

$$Gm_J = 4 * \pi^2 * 10^{-3} * M_J \tag{3.47}$$

berechnen. Als numerisches Verfahren verwenden wir wieder die Verlet-Methode. Die Gleichungen

$$\begin{aligned} x_E(t+\Delta t) &= -x_E(t-\Delta t) + 2x_E(t) - \\ &\quad -\Delta t^2 4\pi^2 \left(\frac{x_E(t)}{r_E^3(t)} + \frac{10^{-3}(x_E(t) - x_J(t)) * M_J}{r_{EJ}^3(t)} \right) \end{aligned} \tag{3.48}$$

$$\begin{aligned} x_J(t+\Delta t) &= -x_J(t-\Delta t) + 2x_J(t) - \\ &\quad -\Delta t^2 4\pi^2 \left(\frac{x_J(t)}{r_J^3(t)} + \frac{3 * 10^{-6}(x_J(t) - x_E(t))}{r_{EJ}^3(t)} \right) \end{aligned} \tag{3.49}$$

gelten für die x-Komponente, die y-Komponente ist analog dazu.

```
   double p0x, p1x, p2x, dp1x, t;
   double p0y, p1y, p2y, dp1y, r3e;
   double q0x, q1x, q2x, dq1x, r3j;
   double q0y, q1y, q2y, dq1y, r3ej;
   double alpha, massj, t_max, dt;
// ...
// loop from t=0 to t=t_max
   t=0.0;
   ofstream outfile("orbit.dat");
   for( int i=0; i<t_max/dt; i++ ) {
        r3e=1./pow(p1x*p1x+p1y*p1y,beta);
        r3j=1./pow(q1x*q1x+q1y*q1y,beta);
        r3ej=1./pow((p1x-q1x)*(p1x-q1x)+(p1y-q1y)*(p1y-q1y),beta);
// Verlet sums
        p2x=2.*p1x-p0x-pi24*dt2*
              (r3e*p1x+r3ej*1.e-3*massj*(p1x-q1x));  // Earth
        p2y=2.*p1y-p0y-pi24*dt2*
              (r3e*p1y+r3ej*1.e-3*massj*(p1y-q1y));
        q2x=2.*q1x-q0x-pi24*dt2*
              (r3j*q1x-r3ej*3.e-6*(p1x-q1x));        // Jupiter
        q2y=2.*q1y-q0y-pi24*dt2*
              (r3j*q1y-r3ej*3.e-6*(p1y-q1y));

        outfile << t << ' ' << p1x << ' ' << p1y <<
                        ' ' << q1x << ' ' << q1y << '\n';
        t=t+dt;
        p0x=p1x;                           // change new->old
        p0y=p1y;
        p1x=p2x;
        p1y=p2y;
        q0x=q1x;                           // change new->old
        q0y=q1y;
```

```
        q1x=q2x;
        q1y=q2y;
    }
```

Abbildung 3.17 zeigt die Bahnen für verschiedene Massen des Jupiters. Gerade in der letzten Grafik ist deutlich zu sehen, daß bei diesem Drei–Körper–Problem die Erde, die deutlich weniger Masse besitzt, als die beiden anderen Himmelskörper, einen chaotischen Verlauf hat. Ebenso ist das allgemeine N–Körper–Problem chaotisch!

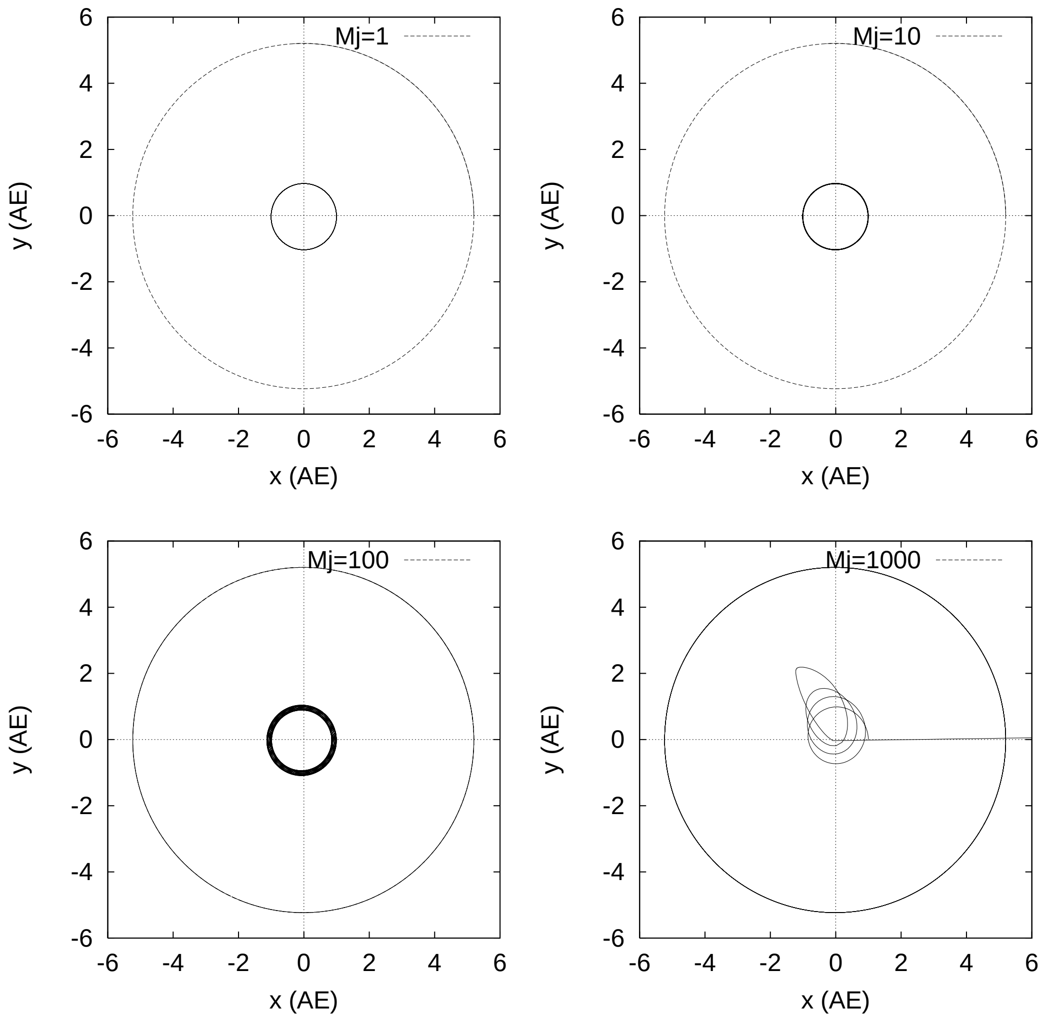


Abbildung 3.17: 3–Körper–Simulation mit verschiedenen Jupiter-Massen.

Die gleiche Methode verwendet man bei Simulation von Molekülen (Molekulardynamik).