

# 重力多体系におけるカオス

柳館佳世子

羽田野研究室

平成 13 年 2 月 28 日

## 概要

重力多体系では、3つの物体だけでカオスになることが知られている。カオスは位置や運動量などの初期値に鋭敏な依存性をもつ。つまり、パラメータによっては初期値が微小に違うだけで全く異なる終状態に行き着く。カオスの判定方法として、リアプノフ指数という量を調べる方法がある。これは、初期値の違いで軌道が時間とともにどのように拡大していくかを示す量であり、カオス状態では正、非カオス状態では負となる。本研究ではコンピュータシミュレーションで3体問題を取り扱い、リアプノフ指数を用いてカオス状態を解析した。

## 目次

1	はじめに	3
2	カオス	3
2.1	カオスとは . . . . .	3
2.2	リアプノフ指数 . . . . .	5
3	3体問題のシミュレーション	6
4	シミュレーション結果	12
5	まとめ	17
6	謝辞	18
A	重力多体系のシミュレーションのプログラム	20
B	3体問題での結果のグラフ	32

# 1 はじめに

日常語としてのカオスとは、無秩序や混乱した状態を表わす言葉のようである。しかし、専門用語のカオスとしては、この表現が適切であるかどうかは疑問である。カオス状態では、その系の未来が予測不可能ではあるが、カオスを構成しているものは決定論的方程式に従っているのである。また、複雑な振舞いを示す系は多体系であり、系自身が複雑なものであると考えがちであるが、実際には異なる3つのパラメータがあればカオス状態になることは可能なのである。

本研究では、重力多体系の中でも特に3体問題を取り扱い、コンピュータシミュレーションによってカオス状態を解析する。

## 2 カオス

### 2.1 カオスとは

#### (i) 微分方程式の解が非周期的になる

カオスとは、決定論に支配される不規則運動のことである。決定論とは「初期条件が決まれば未来永劫にその物体の運動は予測できてしまう」という、出発点を除いては途中で「運不運(偶然性)」の入る余地がないものをいう。それは、ニュートンの運動方程式といった微分方程式であり、そのような方程式を決定論的方程式という。初期条件以外に途中で偶然性の影響を受ける運動は確率過程といわれる。そこでは不規則に訪れる「運不運」はあらかじめその存在と性質が仮定され、もっぱら原因として取扱われるにすぎない。<sup>1)</sup>

カオス系では、決定論的方程式が非決定論的振舞いをする。つまり、カオス系の運動は非周期的であり、系の未来が予測不可能なのである。予測不可能ということは、カオス系は確率系に似ていることは明らかである。しかし、カオスにおける不規則性は系の本質的な力学の一部であり、確率系のような予測できない外からの影響ではない。<sup>2)</sup>このようなカオスは、決定論的カオスと呼ばれる。

カオスが非周期的な解をもつ例として、ローレンツモデルのアトラクターを図1に載せる。図1より、軌跡が1点に収束せず、非周期的になっていることがわかる。

#### (ii) 初期値への鋭敏な依存性をもつ

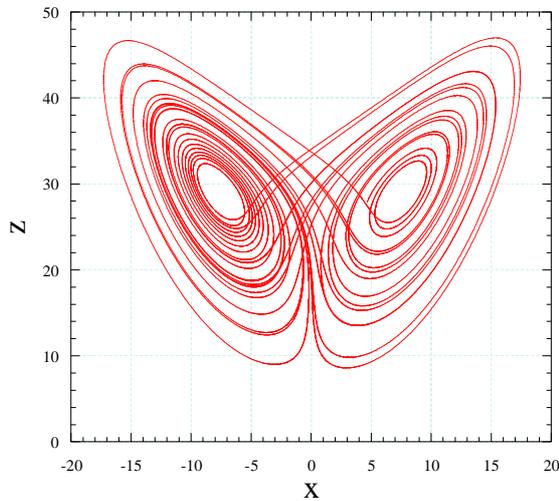


図 1: ローレンツモデルの振舞い ( $x - z$  座標)。<sup>3)</sup>

カオスでは、初期値のほんの微小な違いが時間とともに指数関数的に拡大され、しばらくするうちにまったく違った振舞いとなる。これを概念的に図 2 に示す。左端が始点である。ある初期値を与えた時の 1 点の軌道が青のラインのようになった時、非常に近接した 1 点から出発した軌道が緑のラインのようになったとする。見ればわかるように、2 点の終状態は全く異なっている。さらに、この 2 点間の距離を  $l$  とした時、 $l$  が指数関数的に増大している。このような場合がカオスである。図 2 の場合の 2 点間距離は、位置の差としての距離を示しているが、実際には位置と運動量といった位相空間における距離を指す。

カオスの鋭敏な初期値依存性から分かるように、カオスは計算誤差の影響を非常に受けやすい。計算誤差はこの場合、計算機の精度で決まる。無限の計算過程に、例えば小数点以下 10 桁目に乱れが加わったことと同じである。カオスはこのような微小の乱れの影響を敏感に受ける。これは、自然現象として存在するカオスには大きな意味を持つ。それは、自然界にはいつも乱れが存在するからである。この「決定論的カオス + 微小な乱れ」の系は、本来は決定論的に決まるカオスを、実質的には非決定論的存在としてしまう。これがカオスの現実的側面である。<sup>4)</sup>

また、カオスの予測不可能性は、量子力学の不確定性原理のようにミクロな世界のものではなく、マクロな古典力学が通用する系のものであ

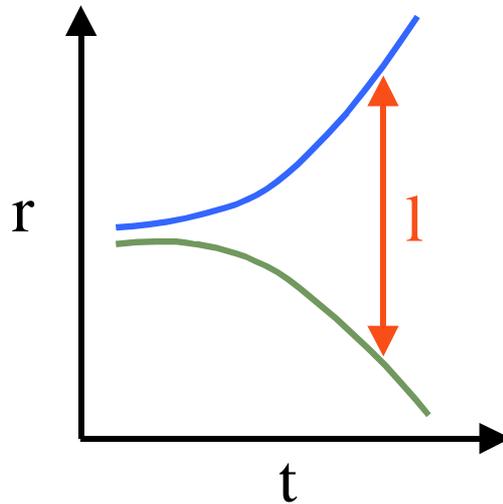


図 2: 鋭敏な初期値依存性。

青:ある初期値での軌跡, 緑:初期値が  $\delta$  だけ違う時の軌跡。

り、実質的な重要性は大きいと考えられる。<sup>4)</sup>

## 2.2 リアプノフ指数

リアプノフ指数とは、わずかに離れた 2 点間の距離の伸縮を定量化する量である。<sup>5)</sup> もっと具体的にいうと、相空間内で十分接近した 2 点から出発した 2 つの軌道が先へ行くにしたがって「乖離していく度合いを示す」量、つまり、カオスの特徴づける重要な量である。<sup>6)</sup>

時刻  $t$  における相空間内の軌道間の距離  $l(t)$  を、

$$l(t) \simeq l(0)e^{\lambda t} \quad (1)$$

で表わす時の  $\lambda$  がリアプノフ指数であり、linear-log の  $t-l$  グラフをかいた時の傾きとなる。ここで、 $l(0)$  は  $t=0$  における軌道間の距離である。

図(3)に  $t-l$  グラフを示した。傾き = リアプノフ指数である。左図は傾き  $> 0$  なのでカオス状態を表わし、右図は傾き  $< 0$  なので非カオス状態を表わしている。

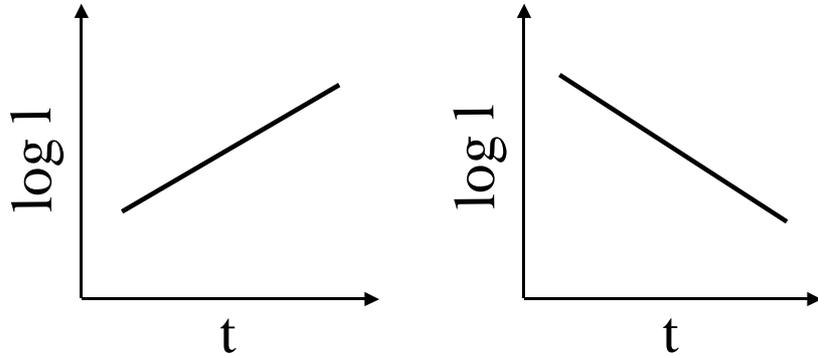


図 3: 左図:カオス状態, 右図:非カオス状態。

### 3 3体問題のシミュレーション

まず、多体問題の取り扱い方を述べる。

初めに運動方程式を求め、そこから位置と運動量の常微分方程式を導出する。その後、4次ルンゲ・クッタ法により、各天体について位置と運動量の近似式  $(x, y, z, p_x, p_y, p_z)$  を求める。また、それぞれの式と  $\Delta$  だけ異なる初期値を持つ近似式  $(x+\Delta x, y+\Delta y, z+\Delta z, p_x+\Delta p_x, p_y+\Delta p_y, p_z+\Delta p_z)$  を求める。天体数を  $n$  とすると、合計  $12n$  個の式の  $T$  秒後の値をコンピュータシミュレーションにより求め、最後にリアプノフ指数を求めるための計算をする。ここでは、3体問題を取り扱ったので、位置  $\vec{r}_i = (x_i, y_i, z_i)$  と運動量  $\vec{p}_i = (p_{xi}, p_{yi}, p_{zi})$  の36次元位相空間の取り扱いになる ( $i = 1, 2, 3$ )。以下、このような流れでさらに詳しく解説する。また、プログラムの流れを図4に示す。

ここでは、3体問題について述べる。3体問題を解くにあたり、まず運動方程式を考える。

$$m_1 \frac{d^2}{dt^2} \vec{r}_2 = \frac{Gm_1m_2}{|\vec{r}_2 - \vec{r}_1|^2} \frac{\vec{r}_2 - \vec{r}_1}{|\vec{r}_2 - \vec{r}_1|} + \frac{Gm_1m_3}{|\vec{r}_3 - \vec{r}_1|^2} \frac{\vec{r}_3 - \vec{r}_1}{|\vec{r}_3 - \vec{r}_1|} \quad (2.1)$$

$$m_2 \frac{d^2}{dt^2} \vec{r}_2 = \frac{Gm_2m_1}{|\vec{r}_1 - \vec{r}_2|^2} \frac{\vec{r}_1 - \vec{r}_2}{|\vec{r}_1 - \vec{r}_2|} + \frac{Gm_2m_3}{|\vec{r}_3 - \vec{r}_2|^2} \frac{\vec{r}_3 - \vec{r}_2}{|\vec{r}_3 - \vec{r}_2|} \quad (2.2)$$

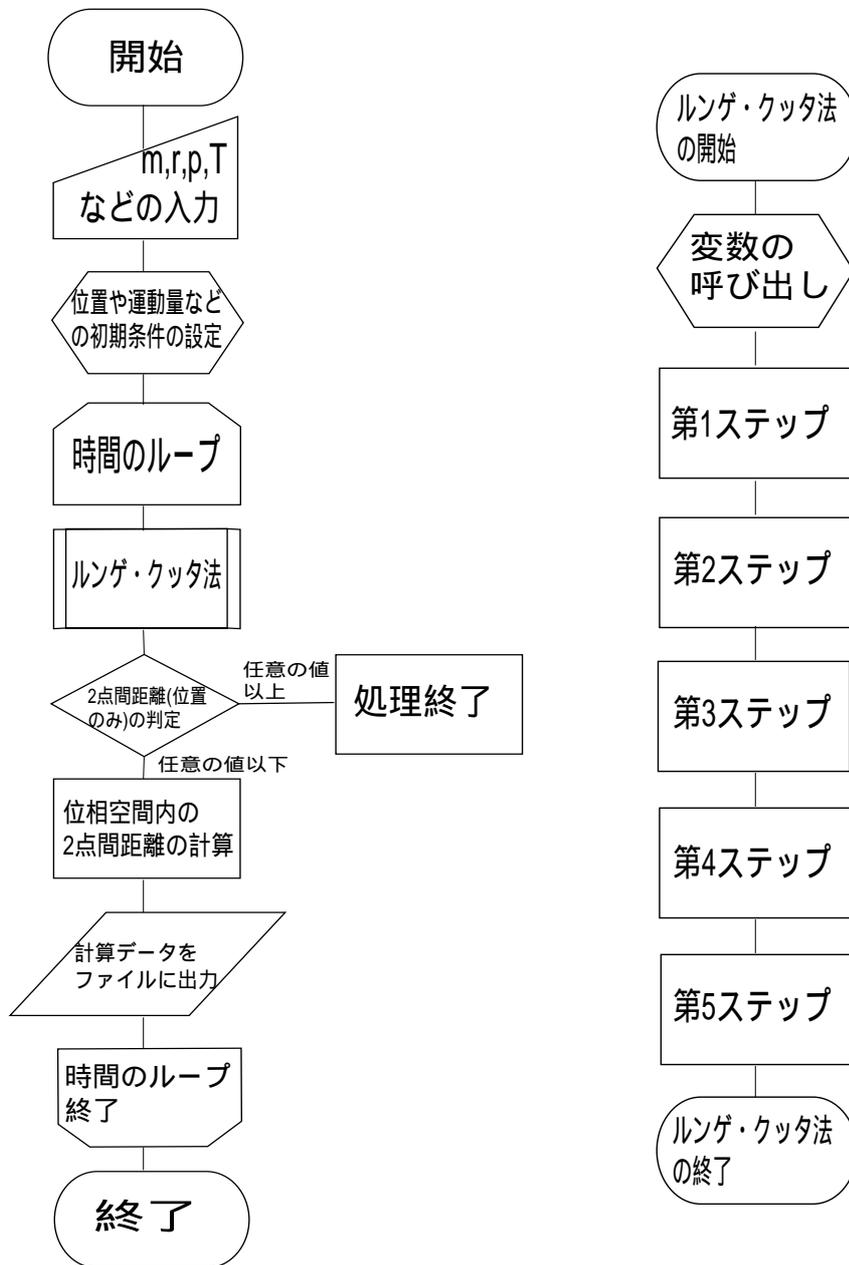


図 4: 左図:全体の流れ図, 右図:Runge・クッタ法の流れ図。

$$m_3 \frac{d^2}{dt^2} \vec{r}_2 = \frac{Gm_3m_2}{|\vec{r}_2 - \vec{r}_3|^2} \frac{\vec{r}_2 - \vec{r}_3}{|\vec{r}_2 - \vec{r}_3|} + \frac{Gm_3m_1}{|\vec{r}_1 - \vec{r}_3|^2} \frac{\vec{r}_1 - \vec{r}_3}{|\vec{r}_1 - \vec{r}_3|} \quad (2.3)$$

ここで、 $G$  は万有引力定数、 $m_i$  は各天体の質量、 $\vec{r}_i$  は各天体の位置である ( $i = 1, 2, 3$ )。式 (2.1) は1つ目の天体、式 (2.2) は2つ目の天体、式 (2.3) は3つ目の天体における万有引力の方程式である。このとき、それぞれの天体は質点として考えることにする。式 (2.1) ,(2.2) ,(2.3) から、位置  $x, y, z$  と運動量  $p_x, p_y, p_z$  の微分方程式を求める。まず、運動量  $\vec{p}$  は、

$$\vec{p} = m \frac{d}{dt} \vec{r} \quad (3)$$

である。つまり、位置  $\vec{r}$  の常微分方程式は、

$$\frac{d}{dt} \vec{r} = \frac{1}{m} \vec{p} \quad (4)$$

である。式 (4) を成分で書くと、

$$\frac{d}{dt} x = \frac{1}{m} p_x \quad (4.1)$$

$$\frac{d}{dt} y = \frac{1}{m} p_y \quad (4.2)$$

$$\frac{d}{dt} z = \frac{1}{m} p_z \quad (4.3)$$

となる。また、運動量  $\vec{p}$  の時間変化は式 (3) より、

$$\frac{d}{dt} \vec{p} = m \frac{d^2}{dt^2} \vec{r} \quad (5.1)$$

で与えられることが分かる。これを成分で書くと、

$$\frac{d}{dt} p_x = m \frac{d^2}{dt^2} x \quad (5.1.1)$$

$$\frac{d}{dt} p_y = m \frac{d^2}{dt^2} y \quad (5.1.2)$$

$$\frac{d}{dt} p_z = m \frac{d^2}{dt^2} z \quad (5.1.3)$$

である。そこで、式 (2.1) と式 (5.1) より、1つ目の天体の運動量  $\vec{p}_1$  の常微分方程式は、

$$\frac{d}{dt}\vec{p}_1 = \frac{Gm_1m_2}{|\vec{r}_2 - \vec{r}_1|^2} \frac{\vec{r}_2 - \vec{r}_1}{|\vec{r}_2 - \vec{r}_1|} + \frac{Gm_1m_3}{|\vec{r}_3 - \vec{r}_1|^2} \frac{\vec{r}_3 - \vec{r}_1}{|\vec{r}_3 - \vec{r}_1|} \quad (5.2)$$

となる。成分で書くと、式 (2.1) と式 (5.1.1),(5.1.2),(5.1.3) より、

$$\frac{d}{dt}p_{x1} = \frac{Gm_1m_2}{|\vec{r}_2 - \vec{r}_1|^2} \frac{x_2 - x_1}{|\vec{r}_2 - \vec{r}_1|} + \frac{Gm_1m_3}{|\vec{r}_3 - \vec{r}_1|^2} \frac{x_3 - x_1}{|\vec{r}_3 - \vec{r}_1|} \quad (5.2.1)$$

$$\frac{d}{dt}p_{y1} = \frac{Gm_1m_2}{|\vec{r}_2 - \vec{r}_1|^2} \frac{y_2 - y_1}{|\vec{r}_2 - \vec{r}_1|} + \frac{Gm_1m_3}{|\vec{r}_3 - \vec{r}_1|^2} \frac{y_3 - y_1}{|\vec{r}_3 - \vec{r}_1|} \quad (5.2.2)$$

$$\frac{d}{dt}p_{z1} = \frac{Gm_1m_2}{|\vec{r}_2 - \vec{r}_1|^2} \frac{z_2 - z_1}{|\vec{r}_2 - \vec{r}_1|} + \frac{Gm_1m_3}{|\vec{r}_3 - \vec{r}_1|^2} \frac{z_3 - z_1}{|\vec{r}_3 - \vec{r}_1|} \quad (5.2.3)$$

となる。2つ目,3つ目の天体についても同様である。

以上の式 (4.1),(4.2),(4.3),(5.2.1),(5.2.2),(5.2.3) は抽象的に6次連立常微分方程式系

$$\frac{d}{dt}x = f_1(x, y, z, p_x) \quad (6.1)$$

$$\frac{d}{dt}y = f_2(x, y, z, p_y) \quad (6.2)$$

$$\frac{d}{dt}z = f_3(x, y, z, p_z) \quad (6.3)$$

$$\frac{d}{dt}p_x = f_4(x, y, z, p_x) \quad (6.4)$$

$$\frac{d}{dt}p_y = f_5(x, y, z, p_y) \quad (6.5)$$

$$\frac{d}{dt}p_z = f_6(x, y, z, p_z) \quad (6.6)$$

の形に書かれている。この連立微分方程式を4次のルンゲ・クッタ法で近似的に解く。4次ルンゲ・クッタ法による数値計算は、以下の5ステップで行われる。

第1ステップ

$$x_a = f_1(x, y, z, p_x)\Delta t \quad (7.1.1)$$

$$y_a = f_2(x, y, z, p_y)\Delta t \quad (7.1.2)$$

$$z_a = f_3(x, y, z, p_z)\Delta t \quad (7.1.3)$$

$$p_{xa} = f_4(x, y, z, p_x)\Delta t \quad (7.1.4)$$

$$p_{ya} = f_5(x, y, z, p_y)\Delta t \quad (7.1.5)$$

$$p_{za} = f_6(x, y, z, p_z)\Delta t \quad (7.1.6)$$

## 第 2 ステップ

$$x_b = f_1\left(x + \frac{x_a}{2}, y + \frac{y_a}{2}, z + \frac{z_a}{2}, p_x + \frac{p_{xa}}{2}\right)\Delta t \quad (7.2.1)$$

$$y_b = f_2\left(x + \frac{x_a}{2}, y + \frac{y_a}{2}, z + \frac{z_a}{2}, p_y + \frac{p_{ya}}{2}\right)\Delta t \quad (7.2.2)$$

$$z_b = f_3\left(x + \frac{x_a}{2}, y + \frac{y_a}{2}, z + \frac{z_a}{2}, p_z + \frac{p_{za}}{2}\right)\Delta t \quad (7.2.3)$$

$$p_{xb} = f_4\left(x + \frac{x_a}{2}, y + \frac{y_a}{2}, z + \frac{z_a}{2}, p_x + \frac{p_{xa}}{2}\right)\Delta t \quad (7.2.4)$$

$$p_{yb} = f_5\left(x + \frac{x_a}{2}, y + \frac{y_a}{2}, z + \frac{z_a}{2}, p_y + \frac{p_{ya}}{2}\right)\Delta t \quad (7.2.5)$$

$$p_{zb} = f_6\left(x + \frac{x_a}{2}, y + \frac{y_a}{2}, z + \frac{z_a}{2}, p_z + \frac{p_{za}}{2}\right)\Delta t \quad (7.2.6)$$

## 第 3 ステップ

$$x_c = f_1\left(x + \frac{x_b}{2}, y + \frac{y_b}{2}, z + \frac{z_b}{2}, p_x + \frac{p_{xb}}{2}\right)\Delta t \quad (7.3.1)$$

$$y_c = f_2\left(x + \frac{x_b}{2}, y + \frac{y_b}{2}, z + \frac{z_b}{2}, p_y + \frac{p_{yb}}{2}\right)\Delta t \quad (7.3.2)$$

$$z_c = f_3\left(x + \frac{x_b}{2}, y + \frac{y_b}{2}, z + \frac{z_b}{2}, p_z + \frac{p_{zb}}{2}\right)\Delta t \quad (7.3.3)$$

$$p_{xc} = f_4\left(x + \frac{x_b}{2}, y + \frac{y_b}{2}, z + \frac{z_b}{2}, p_x + \frac{p_{xb}}{2}\right)\Delta t \quad (7.3.4)$$

$$p_{yc} = f_5\left(x + \frac{x_b}{2}, y + \frac{y_b}{2}, z + \frac{z_b}{2}, p_y + \frac{p_{yb}}{2}\right)\Delta t \quad (7.3.5)$$

$$p_{zc} = f_6\left(x + \frac{x_b}{2}, y + \frac{y_b}{2}, z + \frac{z_b}{2}, p_z + \frac{p_{zb}}{2}\right)\Delta t \quad (7.3.6)$$

#### 第4ステップ

$$x_d = f_1(x + x_c, y + y_c, z + z_c, p_x + p_{xc})\Delta t \quad (7.4.1)$$

$$y_d = f_2(x + x_c, y + y_c, z + z_c, p_y + p_{yc})\Delta t \quad (7.4.2)$$

$$z_d = f_3(x + x_c, y + y_c, z + z_c, p_z + p_{zc})\Delta t \quad (7.4.3)$$

$$p_{xd} = f_4(x + x_c, y + y_c, z + z_c, p_x + p_{xc})\Delta t \quad (7.4.4)$$

$$p_{yd} = f_5(x + x_c, y + y_c, z + z_c, p_y + p_{yc})\Delta t \quad (7.4.5)$$

$$p_{zd} = f_6(x + x_c, y + y_c, z + z_c, p_z + p_{zc})\Delta t \quad (7.4.6)$$

#### 第5ステップ

$$x_{\text{new}} = x + (x_a + 2x_b + 2x_c + x_d)/6 \quad (8.1)$$

$$y_{\text{new}} = y + (y_a + 2y_b + 2y_c + y_d)/6 \quad (8.2)$$

$$z_{\text{new}} = z + (z_a + 2z_b + 2z_c + z_d)/6 \quad (8.3)$$

$$p_{x\text{new}} = p_x + (p_{xa} + 2p_{xb} + 2p_{xc} + p_{xd})/6 \quad (8.4)$$

$$p_{y\text{new}} = p_y + (p_{ya} + 2p_{yb} + 2p_{yc} + p_{yd})/6 \quad (8.5)$$

$$p_{z\text{new}} = p_z + (p_{za} + 2p_{zb} + 2p_{zc} + p_{zd})/6 \quad (8.6)$$

式 (8.1),(8.2),(8.3),(8.4),(8.5),(8.6) は、それぞれの  $\Delta t$  秒後の値、つまり  $n + 1$  番目を表わす近似式である。<sup>7)</sup>

次に、各式と  $\Delta$  だけ異なる値を持つ近似式を出す。 $(x, y, z, px, py, pz)$  に  $(x + \Delta x, y + \Delta y, z + \Delta z, px + \Delta px, py + \Delta py, pz + \Delta pz)$  を代入し、同様に 4 次ルンゲ・クッタ法を用いて求める。

最後に、リアプノフ指数を調べるために、 $6n$  次元位相空間中の 2 点間の距離  $l$  を計算する。 $12n$  個の式の  $T$  秒後の値を、それぞれ  $(X_i, Y_i, Z_i, Px_i, Py_i, Pz_i)$ ,  $(X'_i, Y'_i, Z'_i, P'x'_i, P'y'_i, P'z'_i)$  とする (今の場合、 $n = 3, i = 1, 2, 3$ ) と、 $l$  は、

$$l = \sqrt{\frac{(X_1 - X'_1)^2 + (Y_1 - Y'_1)^2 + (Z_1 - Z'_1)^2}{(Px_1 - P'x'_1)^2 + (Py_1 - P'y'_1)^2 + (Pz_1 - P'z'_1)^2} + \frac{(X_2 - X'_2)^2 + (Y_2 - Y'_2)^2 + (Z_2 - Z'_2)^2}{(Px_2 - P'x'_2)^2 + (Py_2 - P'y'_2)^2 + (Pz_2 - P'z'_2)^2} + \frac{(X_3 - X'_3)^2 + (Y_3 - Y'_3)^2 + (Z_3 - Z'_3)^2}{(Px_3 - P'x'_3)^2 + (Py_3 - P'y'_3)^2 + (Pz_3 - P'z'_3)^2}} \quad (9)$$

となる。

以上の計算を行うプログラムは後半に載せる。また、3 体問題の時には初期値は手で与えたが、それ以上の天体数の時には乱数で与えるようにした。載せたプログラムは、この場合のものである。3 体問題における初期値は次節に表記する。

## 4 シミュレーション結果

3 体問題でカオスとなった時の初期値を以下に示す。

$$\begin{aligned} m_1 &= 0.8256 \\ m_2 &= 1.489089 \\ m_3 &= 1.14507 \\ x_1 &= 1.0 \\ y_1 &= 0.0 \\ z_1 &= 0.0 \\ p_{x1} &= 0.095712 \\ p_{y1} &= 0.326 \end{aligned}$$

$$\begin{aligned}
p_{z1} &= -0.01 \\
x_2 &= -0.5 \\
y_2 &= 0.866 \\
z_2 &= 0.0 \\
p_{x2} &= -1.9758 \\
p_{y2} &= -0.479 \\
p_{z2} &= -0.523 \\
x_3 &= -0.5 \\
y_3 &= -0.866 \\
z_3 &= 0.0 \\
p_{x3} &= 0.8561 \\
p_{y3} &= -0.325 \\
p_{z3} &= 0.6301 \\
T &= 250.0 \\
dt &= 0.00001 \\
\delta &= 0.01
\end{aligned}$$

$m_i$  は各天体の質量、 $x_i, y_i, z_i$  は各天体の位置、 $p_{xi}, p_{yi}, p_{zi}$  は各天体の運動量、 $T$  は時間、 $dt$  は時間の刻み幅、 $\delta$  は初期値からのずれである ( $i=1,2,3$ )。また、このシミュレーションでは万有引力定数  $G$  を 1.0 としている。

図 5 に、初めに与えた初期値をもつ 1 つ目の天体の  $x - y$  座標の軌跡と、その初期値と  $\delta$  だけ異なる初期値をもつ天体の  $x - y$  座標の軌跡を示した。右上が始点である。ここから、初期値に鋭敏な依存性をもっていることがわかる。

図 6 に、3 天体それぞれの  $x - y$  座標の軌跡を示した。右上が始点である。ここから、すべての天体の運動が非周期的になっていることがわかる。

図 7 に、linear-log の  $t - l$  グラフを示した。ここから、右上がりの直線が得られていることがわかる。(初めの方は初期状態の影響が残っている初期緩和だと考えられるので無視する。) この図のリアプノフ指数はおおよそ  $\lambda = 0.013$  である。

以上の結果から、3 体問題におけるシミュレーションでカオス状態とな

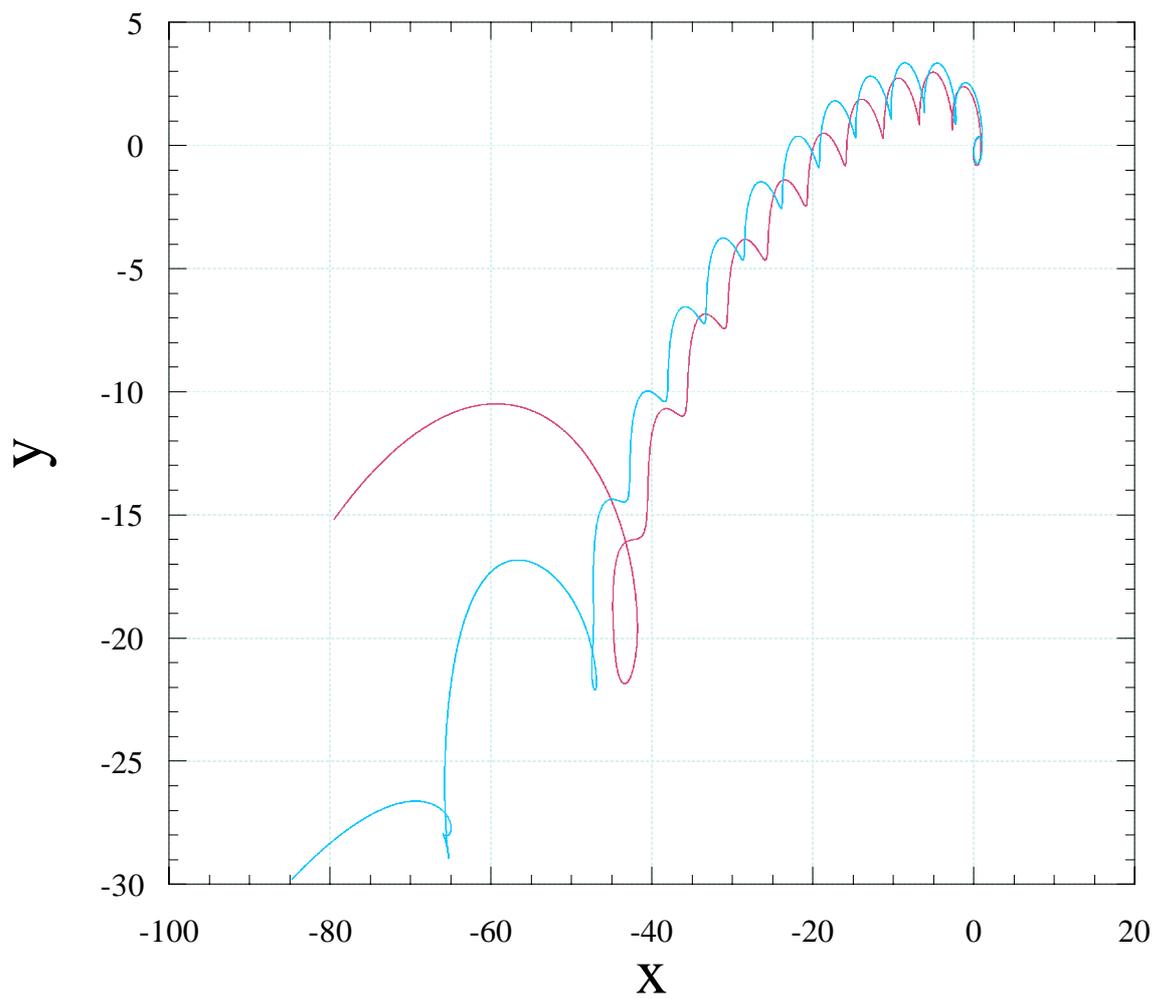


図 5: 1つ目の天体の  $x - y$  座標。  
 赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

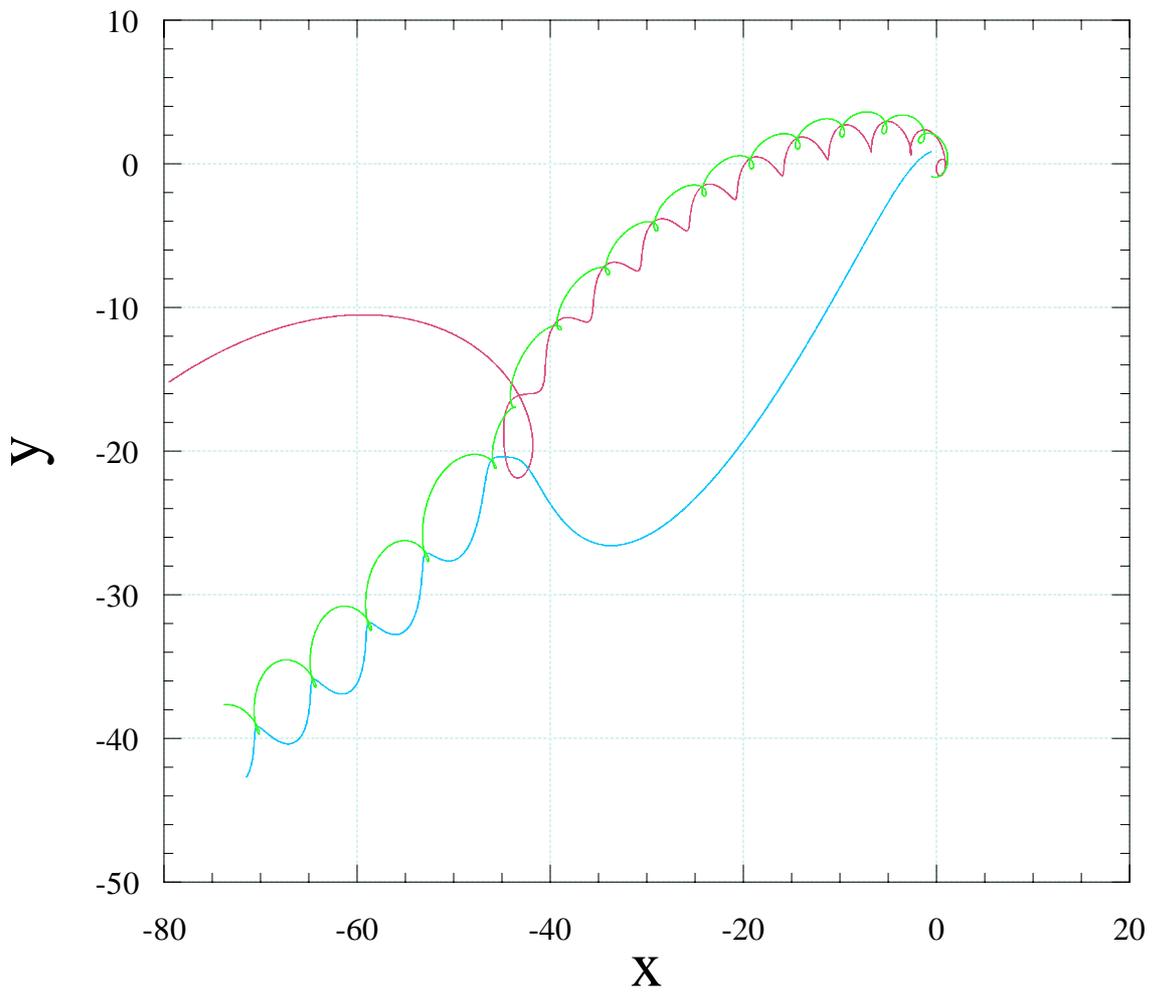


図 6: 3 天体の軌跡 ( $x - y$  座標)。  
 赤:1 つ目の天体, 青:2 つ目の天体, 緑:3 つ目の天体。

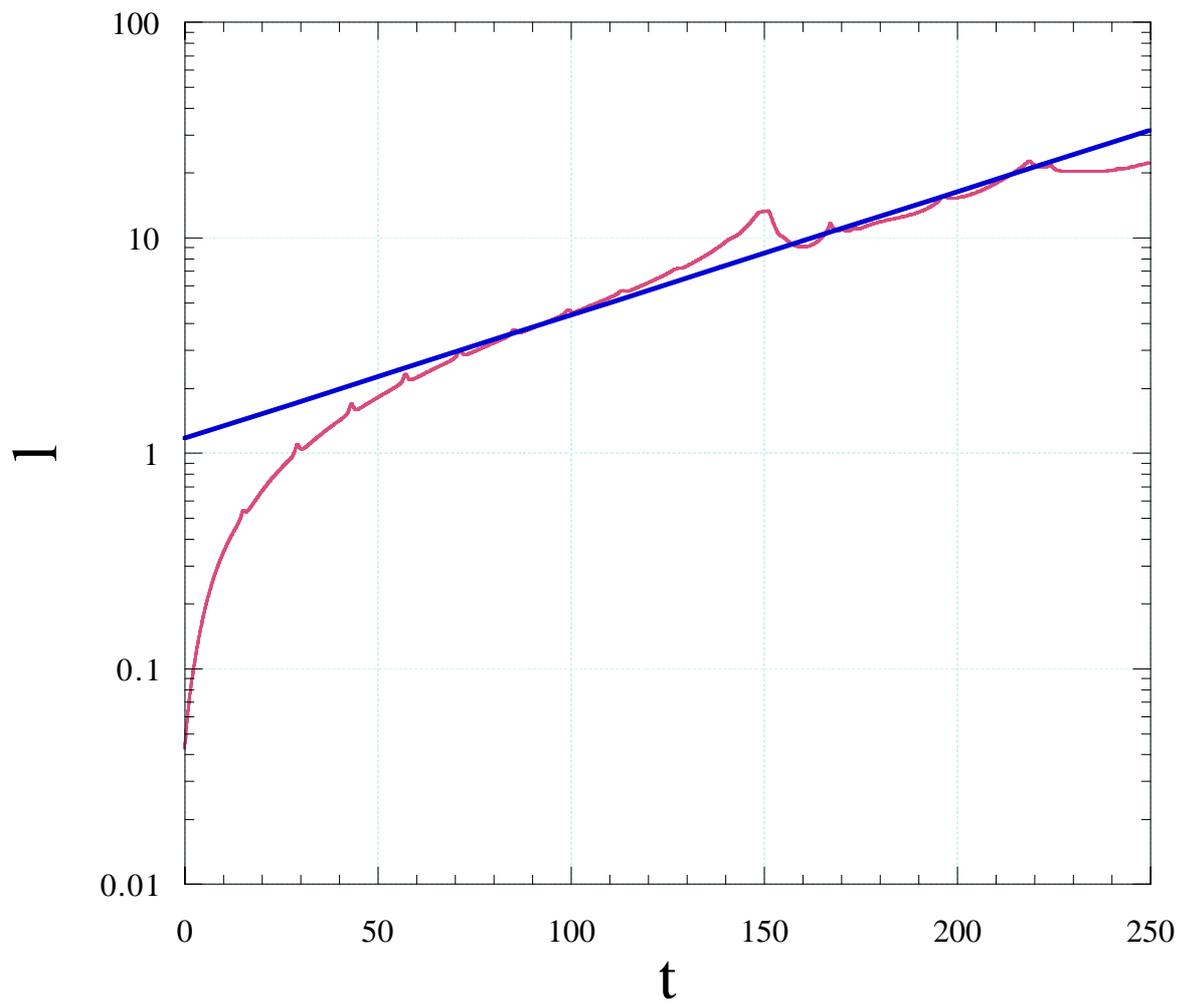


図 7:  $t-l$ (linear-log) グラフ。  
赤: $t-l$  プロット, 青:傾きの近似直線。

ることを確認できた。

なお、他の軸をプロットした図は最後に載せておく。

## 5 まとめ

ここまでで、重力多体系において3体問題でカオスになることを確認した。微分方程式の近似方法は4次ルンゲ・クッタ法を用い、時間の刻み幅は $10^{-5}$ にとっている。また、4次ルンゲ・クッタ法を用いる前に、2次の近似方法であるヴェルレ法を用いたプログラムを使用していたが、今回与えた初期値での振舞いにおいて、全く同じ状態が得られているので、ヴェルレ法より格段に精度の良い近似方法を用いている現在において、シミュレーションの精度には問題がないと考えられる。よって、このプログラムで天体数を増やしたシミュレーションを行うことは可能であると考えられる。

そこで、初期値を与える方法を乱数で与えるように変更してシミュレーションを行うことにした。しかし、カオス状態になったという結果は一度も得ることができなかった。多くの場合、非周期的な運動にならずに各天体がそれぞれバラバラな方向に直線的に飛んでいってしまうのである。一番カオスに近い状態であっても、非周期的な振舞いをするのは初期のみで、すぐにバラバラになってしまう。

前述したように、3体問題に対する結果を見る限り、プログラム自身には何も問題はないと思われる。よって、考えられることは初期値の問題である。乱数を与える初期値の範囲を微小に変化させただけで、全く異なる軌道を得ることができる。天体数が一定の時にはもちろん、天体数を変えるごとに系の状態は異なってくる。つまり、カオスになる初期値が存在することは確実であるが、その値を見つけることは容易ではない。

ただ、非常に多くの重力多体系(例えば、星団など)において、初めにあった天体すべてがカオス的振舞いをするとは考えにくい。よって、任意のある一定距離以上離れてしまった天体については無視するようにプログラミングすることにした。また、銀河の中心には巨大なブラックホールが存在すると考えられているように、非常に多くの天体がある空間内に留まり続けるためには、そのすべてを引き付ける質量を持った天体が必要であるとも考えられるので、中心に初期条件で運動量 = 0 の、質量が非常に大きい天体を置くことにした。後半に載せてあるプログラムはこのようにして作ったものである。よって、初期値の範囲さえ見つける

ことができれば、このプログラムでさらに多くの重力多体問題でカオスの解析ができると考えられる。

## 6 謝辞

2年間、特にこの研究をした1年間、親切丁寧に指導して下さった羽田野直道先生に深い尊敬と感謝を致しております。また、この研究に適切なアドバイスをして下さった先生方、同研究室の大学院生の方々、学部研究生と友人の皆さんには大変お世話になりました。この場を借りて御礼申し上げます。

## 参考文献

- 1) 竹山協三 『カオス 自然の乱れ方』 (裳華房)
- 2) G・L・ベイカー、J・P・ゴラブ 著, 松下貢 訳 『カオス力学入門 基礎とシミュレーション』 (啓学出版)
- 3) 藤渕智康 『ローレンツモデルにおけるカオスの定量的解析』 (青山学院大学 2000 年度卒業論文)
- 4) 長島弘幸、馬場良和 『カオス入門 現象の解析と数理』 (培風館)
- 5) 高安秀樹 『フラクタル』 (朝倉書店)
- 6) 早間慧 『カオス力学の基礎』 (現代数学社)
- 7) 下條隆嗣 『カオス力学入門』 (近代科学社)

## A 重力多体系のシミュレーションのプログラム

```
#include<stdio.h>
#include<math.h>
#include<string.h>
#include<stdlib.h>
#define p 4 /* star's number +1 */

struct star{
    double m;
    double x;   double y;   double z;
    double px;  double py;  double pz;
    int flag;
};

void runge(struct star st[],struct star n_st[],double *Gp,float *dtp,
           double *sumxp,double *sumyp,double *sumzp,int *Kp);

main()
{
    FILE *fp[p],*fpd[p],*fl;
    char filename1[p][30],filename2[p][30],filenumber[30],filetail[30];
    float t,T,dt,d,L;
    int N,it,n,i,j,K;
    double G,l;
    double distance,distance_d;
    double m,x,y,z,px,py,pz,R,P;
    float dx,dy,dz,dpx,dpy,dpz;
    double sumx,sumy,sumz,sumxd,sumyd,sumzd;
    struct star st[p],std[p],n_st[p],n_std[p];

    G = 1.0;

    K = p-1;
```

```

printf("st[0].m = std[0].m = ");
scanf("%lf",&st[0].m);
std[0].m = st[0].m;
st[0].x = 0.0;   st[0].y = 0.0;   st[0].z = 0.0;
st[0].px = 0.0; st[0].py = 0.0; st[0].pz = 0.0;

printf("0 <= st.m <= m\nm = ");
scanf("%lf",&m);
printf("-x <= st.x <= x\nx = ");
scanf("%lf",&x);
printf("-y <= st.y <= y\ny = ");
scanf("%lf",&y);
printf("-z <= st.z <= z\nz = ");
scanf("%lf",&z);
printf("-px <= st.px <= px\npx = ");
scanf("%lf",&px);
printf("-py <= st.py <= py\npy = ");
scanf("%lf",&py);
printf("-pz <= st.pz <= pz\npz = ");
scanf("%lf",&pz);
printf("T = ");
scanf("%f",&T);
printf("dt = ");
scanf("%f",&dt);
printf("delta = ");
scanf("%f",&d);
n = 2000;
printf("effective region = ");
scanf("%f",&L);

R = sqrt(x*x+y*y+z*z);
P = sqrt(px*px+py*py+pz*pz);

for(i=1; i<=K; i++)
{

```

```

    st[i].m = std[i].m = ((double)rand()/RAND_MAX)*m;

do{
    st[i].x = ((double)rand()/RAND_MAX)*2*x-x;
    st[i].y = ((double)rand()/RAND_MAX)*2*y-y;
    st[i].z = ((double)rand()/RAND_MAX)*2*z-z;
}while(sqrt(st[i].x*st[i].x+st[i].y*st[i].y+st[i].z*st[i].z) > R);

do{
    st[i].px = ((double)rand()/RAND_MAX)*2*px-px;
    st[i].py = ((double)rand()/RAND_MAX)*2*py-py;
    st[i].pz = ((double)rand()/RAND_MAX)*2*pz-pz;
}while(sqrt(st[i].px*st[i].px+st[i].py*st[i].py+st[i].pz*st[i].pz) > P);

st[i].flag = 0;
}

dx = dy = dz = dpx = dpy = dpz = d;

for(i=0; i<=K; i++)
{
    std[i].x=st[i].x+dx;
    std[i].y=st[i].y+dy;
    std[i].z=st[i].z+dz;
    std[i].px=st[i].px+dpx;
    std[i].py=st[i].py+dpy;
    std[i].pz=st[i].pz+dpz;
}

strcpy(filetail,"_1.dat");
for(i=0; i<=K; i++)
{
    strcpy(filename1[i],"st");

```

```

strcpy(filename2[i],"std");
sprintf(filename1,"%03d",i);
strcat(filename1,filename2[i]);
strcat(filename2[i],filename1);
strcat(filename1,filetail);
strcat(filename2[i],filetail);

fp[i]=fopen(filename1[i],"w");
fpd[i]=fopen(filename2[i],"w");
}
fl=fopen("stl_1.dat","w");

N=T/dt;

for(it=0; it<=N; it=it+1)
{
    t=dt*it;

    l=0.0;

    runge(st,n_st,&G,&dt,&sumx,&sumy,&sumz,&K);
    runge(std,n_std,&G,&dt,&sumxd,&sumyd,&sumzd,&K);

    for(i=0; i<=K; i++)
    {
        st[i].x=n_st[i].x;    std[i].x=n_std[i].x;
        st[i].y=n_st[i].y;    std[i].y=n_std[i].y;
        st[i].z=n_st[i].z;    std[i].z=n_std[i].z;
        st[i].px=n_st[i].px;  std[i].px=n_std[i].px;
        st[i].py=n_st[i].py;  std[i].py=n_std[i].py;
        st[i].pz=n_st[i].pz;  std[i].pz=n_std[i].pz;
    }
}

```

```

distance = sqrt(st[i].x*st[i].x+st[i].y*st[i].y+st[i].z*st[i].z);
distance_d = sqrt(std[i].x*std[i].x+std[i].y*std[i].y
                +std[i].z*std[i].z);

if((distance > L || distance_d > L) && i != 0)
{
    st[i].flag = 1;
}

if(st[i].flag == 0)
{
    l += ((st[i].x-std[i].x)*(st[i].x-std[i].x))
        +((st[i].y-std[i].y)*(st[i].y-std[i].y))
        +((st[i].z-std[i].z)*(st[i].z-std[i].z))
        +((st[i].px-std[i].px)*(st[i].px-std[i].px))
        +((st[i].py-std[i].py)*(st[i].py-std[i].py))
        +((st[i].pz-std[i].pz)*(st[i].pz-std[i].pz));
}
}

l = sqrt(l);

if(it%n==0 || it%n==1)
{
    for(i=0; i<=K; i++)
    {
        if(st[i].flag == 0)
        {
            fprintf(fp[i],"%e %e %e %e %e %e %e\n"
                    ,t,st[i].x,st[i].y,st[i].z,st[i].px,st[i].py,st[i].pz);
            fprintf(fpd[i],"%e %e %e %e %e %e %e\n"
                    ,t,std[i].x,std[i].y,std[i].z,
                    std[i].px,std[i].py,std[i].pz);
        }
    }
}

```

```

        }
    }
    fprintf(fl,"%e %15.121f\n",t,l);
}
}

for(i=0; i<=K; i++)
{
    fclose(fp[i]);
    fclose(fpd[i]);
}
fclose(fl);
}

```

```

void runge(struct star st[],struct star n_st[],double *Gp,float *dtp,
           double *sumxp,double *sumyp,double *sumzp,int *Kp)
{
    int i,j;
    struct star st_a[p],st_b[p],st_c[p],st_d[p],str[p];
    double r;

    for(i=0; i<=*Kp; i++)
    {
        *sumxp=0.0;   *sumyp=0.0;   *sumzp=0.0;

        if(st[i].flag == 0)
        {
            for(j=0; j<=*Kp; j++)
            {

                if((j != i) && (st[j].flag == 0))
                {

```

```

        r = (st[j].x-st[i].x)*(st[j].x-st[i].x)
            +(st[j].y-st[i].y)*(st[j].y-st[i].y)
            +(st[j].z-st[i].z)*(st[j].z-st[i].z);

        *sumxp += (st[j].m*(st[j].x-st[i].x))/(r*sqrt(r));
        *sumyp += (st[j].m*(st[j].y-st[i].y))/(r*sqrt(r));
        *sumzp += (st[j].m*(st[j].z-st[i].z))/(r*sqrt(r));

    }
}

st_a[i].x=(st[i].px/st[i].m)*(*dtp);
st_a[i].y=(st[i].py/st[i].m)*(*dtp);
st_a[i].z=(st[i].pz/st[i].m)*(*dtp);

st_a[i].px>(*Gp)*(st[i].m)*(*sumxp)*(*dtp);
st_a[i].py>(*Gp)*(st[i].m)*(*sumyp)*(*dtp);
st_a[i].pz>(*Gp)*(st[i].m)*(*sumzp)*(*dtp);

}
}

for(i=0; i<=*Kp; i++)
{
    if(st[i].flag == 0)
    {
        str[i].x=st[i].x+(st_a[i].x/2);
        str[i].y=st[i].y+(st_a[i].y/2);
        str[i].z=st[i].z+(st_a[i].z/2);
        str[i].px=st[i].px+(st_a[i].px/2);
        str[i].py=st[i].py+(st_a[i].py/2);
        str[i].pz=st[i].pz+(st_a[i].pz/2);
    }
}
}

```

```

for(i=0; i<=*Kp; i++)
{
    *sumxp=0.0;    *sumyp=0.0;    *sumzp=0.0;

    if(st[i].flag == 0)
    {
        for(j=0; j<=*Kp; j++)
        {

            if((j != i) && (st[j].flag == 0))
            {

                r = (str[j].x-str[i].x)*(str[j].x-str[i].x)
                    +(str[j].y-str[i].y)*(str[j].y-str[i].y)
                    +(str[j].z-str[i].z)*(str[j].z-str[i].z);

                *sumxp += (st[j].m*(str[j].x-str[i].x))/(r*sqrt(r));
                *sumyp += (st[j].m*(str[j].y-str[i].y))/(r*sqrt(r));
                *sumzp += (st[j].m*(str[j].z-str[i].z))/(r*sqrt(r));

            }
        }

        st_b[i].x=(str[i].px/st[i].m)*(*dtp);
        st_b[i].y=(str[i].py/st[i].m)*(*dtp);
        st_b[i].z=(str[i].pz/st[i].m)*(*dtp);

        st_b[i].px>(*Gp)*(st[i].m)*(*sumxp)*(*dtp);
        st_b[i].py>(*Gp)*(st[i].m)*(*sumyp)*(*dtp);
        st_b[i].pz>(*Gp)*(st[i].m)*(*sumzp)*(*dtp);
    }
}

for(i=0; i<=*Kp; i++)
{

```

```

if(st[i].flag == 0)
{
    str[i].x=st[i].x+(st_b[i].x/2);
    str[i].y=st[i].y+(st_b[i].y/2);
    str[i].z=st[i].z+(st_b[i].z/2);
    str[i].px=st[i].px+(st_b[i].px/2);
    str[i].py=st[i].py+(st_b[i].py/2);
    str[i].pz=st[i].pz+(st_b[i].pz/2);
}
}

for(i=0; i<=*Kp; i++)
{
    *sumxp=0.0;    *sumyp=0.0;    *sumzp=0.0;

if(st[i].flag == 0)
{
    for(j=0; j<=*Kp; j++)
    {

if((j != i) && (st[j].flag == 0))
{

    r = (str[j].x-str[i].x)*(str[j].x-str[i].x)
        +(str[j].y-str[i].y)*(str[j].y-str[i].y)
        +(str[j].z-str[i].z)*(str[j].z-str[i].z);

    *sumxp += (st[j].m*(str[j].x-str[i].x))/(r*sqrt(r));
    *sumyp += (st[j].m*(str[j].y-str[i].y))/(r*sqrt(r));
    *sumzp += (st[j].m*(str[j].z-str[i].z))/(r*sqrt(r));

}
}

st_c[i].x=(str[i].px/st[i].m)*(*dtp);

```

```

st_c[i].y=(str[i].py/st[i].m)*(*dtp);
st_c[i].z=(str[i].pz/st[i].m)*(*dtp);

st_c[i].px>(*Gp)*(st[i].m)*(*sumxp)*(*dtp);
st_c[i].py>(*Gp)*(st[i].m)*(*sumyp)*(*dtp);
st_c[i].pz>(*Gp)*(st[i].m)*(*sumzp)*(*dtp);

}
}

for(i=0; i<=*Kp; i++)
{
if(st[i].flag == 0)
{
str[i].x=st[i].x+st_c[i].x;
str[i].y=st[i].y+st_c[i].y;
str[i].z=st[i].z+st_c[i].z;
str[i].px=st[i].px+st_c[i].px;
str[i].py=st[i].py+st_c[i].py;
str[i].pz=st[i].pz+st_c[i].pz;
}
}

for(i=0; i<=*Kp; i++)
{
*sumxp=0.0; *sumyp=0.0; *sumzp=0.0;

if(st[i].flag == 0)
{
for(j=0; j<=*Kp; j++)
{

if((j != i) && (st[j].flag == 0))
{

```

```

        r = (str[j].x-str[i].x)*(str[j].x-str[i].x)
            +(str[j].y-str[i].y)*(str[j].y-str[i].y)
            +(str[j].z-str[i].z)*(str[j].z-str[i].z);

        *sumxp += (st[j].m*(str[j].x-str[i].x))/(r*sqrt(r));
        *sumyp += (st[j].m*(str[j].y-str[i].y))/(r*sqrt(r));
        *sumzp += (st[j].m*(str[j].z-str[i].z))/(r*sqrt(r));

    }
}

st_d[i].x=(str[i].px/st[i].m)*(*dtp);
st_d[i].y=(str[i].py/st[i].m)*(*dtp);
st_d[i].z=(str[i].pz/st[i].m)*(*dtp);

st_d[i].px>(*Gp)*(st[i].m)*(*sumxp)*(*dtp);
st_d[i].py>(*Gp)*(st[i].m)*(*sumyp)*(*dtp);
st_d[i].pz>(*Gp)*(st[i].m)*(*sumzp)*(*dtp);

}
}

for(i=0; i<=*Kp; i++)
{
if(st[i].flag == 0)
{
    n_st[i].x=st[i].x+(st_a[i].x+2*st_b[i].x
                    +2*st_c[i].x+st_d[i].x)/6;
    n_st[i].y=st[i].y+(st_a[i].y+2*st_b[i].y
                    +2*st_c[i].y+st_d[i].y)/6;
    n_st[i].z=st[i].z+(st_a[i].z+2*st_b[i].z
                    +2*st_c[i].z+st_d[i].z)/6;
    n_st[i].px=st[i].px+(st_a[i].px+2*st_b[i].px
                    +2*st_c[i].px+st_d[i].px)/6;

```

```
n_st[i].py=st[i].py+(st_a[i].py+2*st_b[i].py
                +2*st_c[i].py+st_d[i].py)/6;
n_st[i].pz=st[i].pz+(st_a[i].pz+2*st_b[i].pz
                +2*st_c[i].pz+st_d[i].pz)/6;
}
}
}
```

## B 3体問題での結果のグラフ

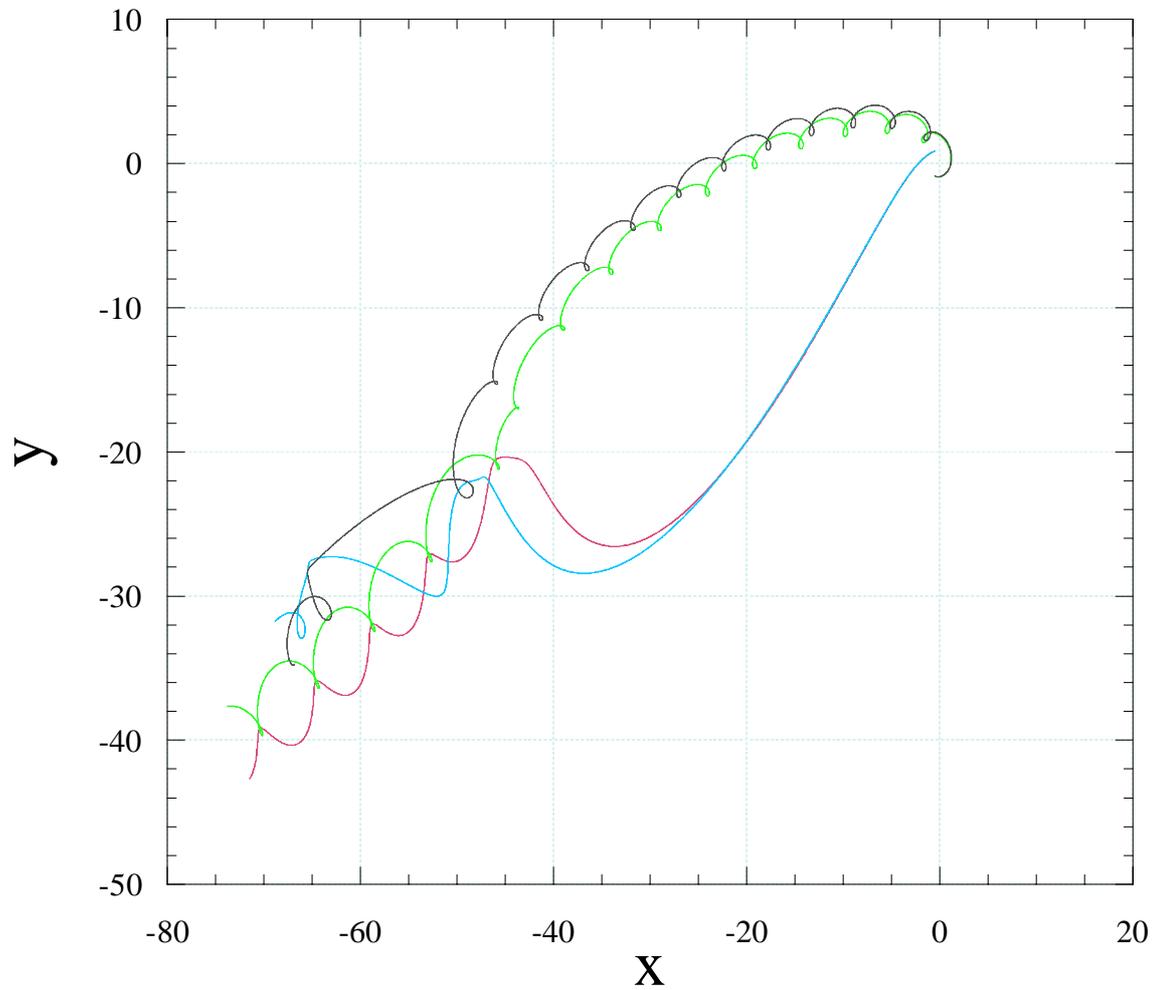


図 8: 2つ目と3つ目の天体の  $x - y$  座標。

赤:初めに与えた初期値での軌跡 (2つ目の天体), 青:初期値が  $\delta$  だけ異なる時の軌跡 (2つ目の天体),  
緑:初めに与えた初期値での軌跡 (3つ目の天体), 黒:初期値が  $\delta$  だけ異なる時の軌跡 (3つ目の天体)。

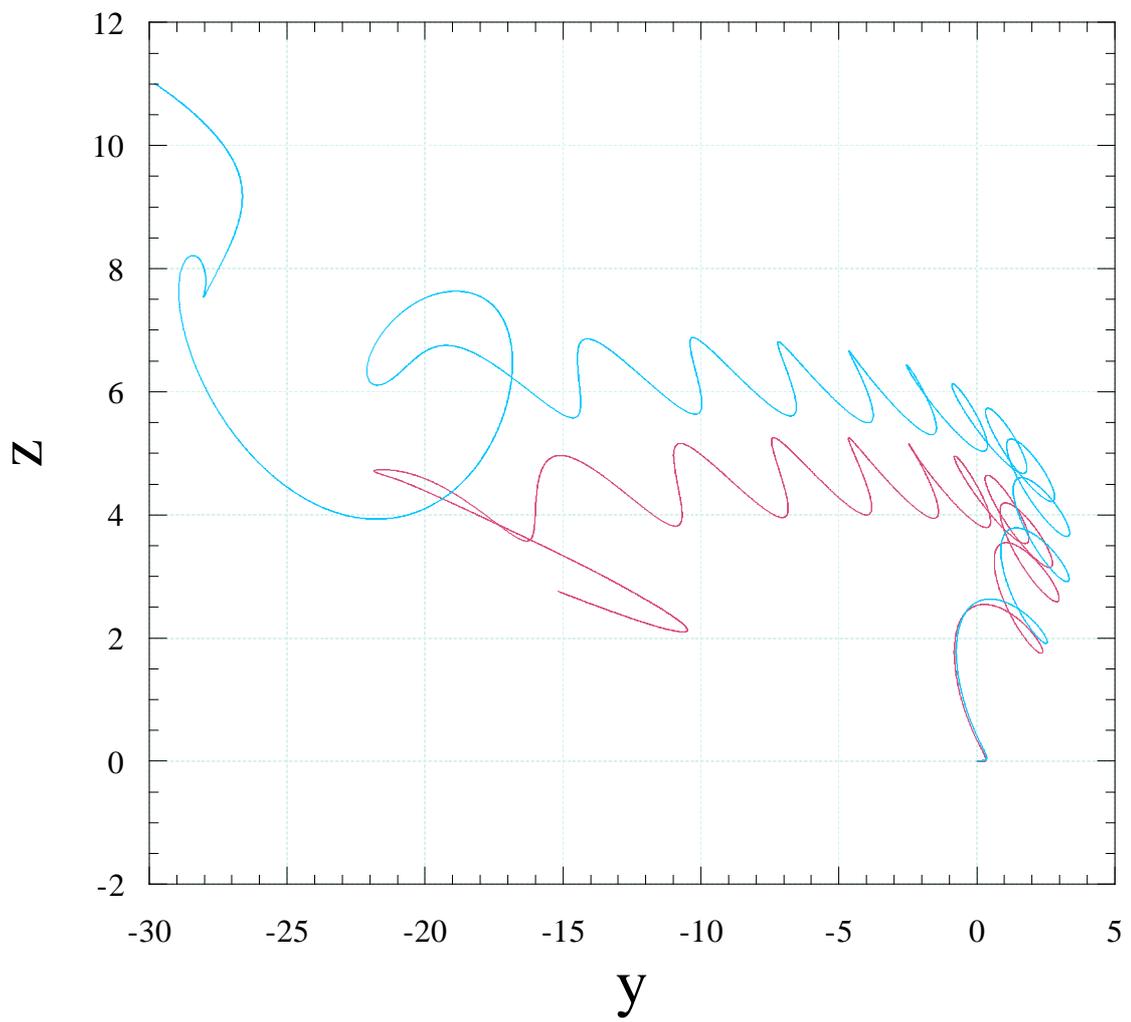


図 9: 1 つ目の天体の  $y - z$  座標。  
 赤: 初めに与えた初期値での軌跡, 青: 初期値が  $\delta$  だけ異なる時の軌跡。

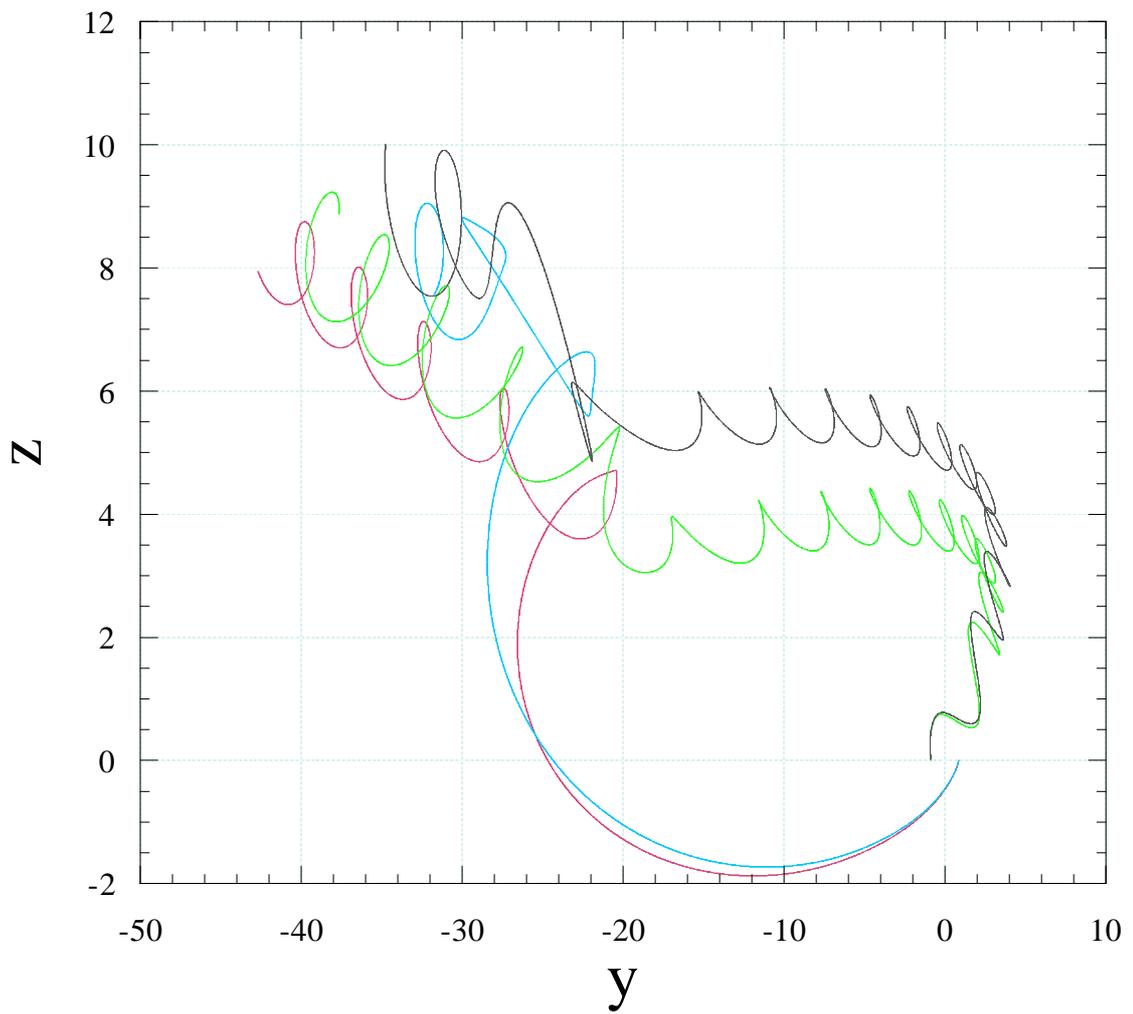


図 10: 2つ目と3つ目の天体の  $y - z$  座標。

赤:初めに与えた初期値での軌跡 (2つ目の天体), 青:初期値が  $\delta$  だけ異なる時の軌跡 (2つ目の天体),  
 緑:初めに与えた初期値での軌跡 (3つ目の天体), 黒:初期値が  $\delta$  だけ異なる時の軌跡 (3つ目の天体)。

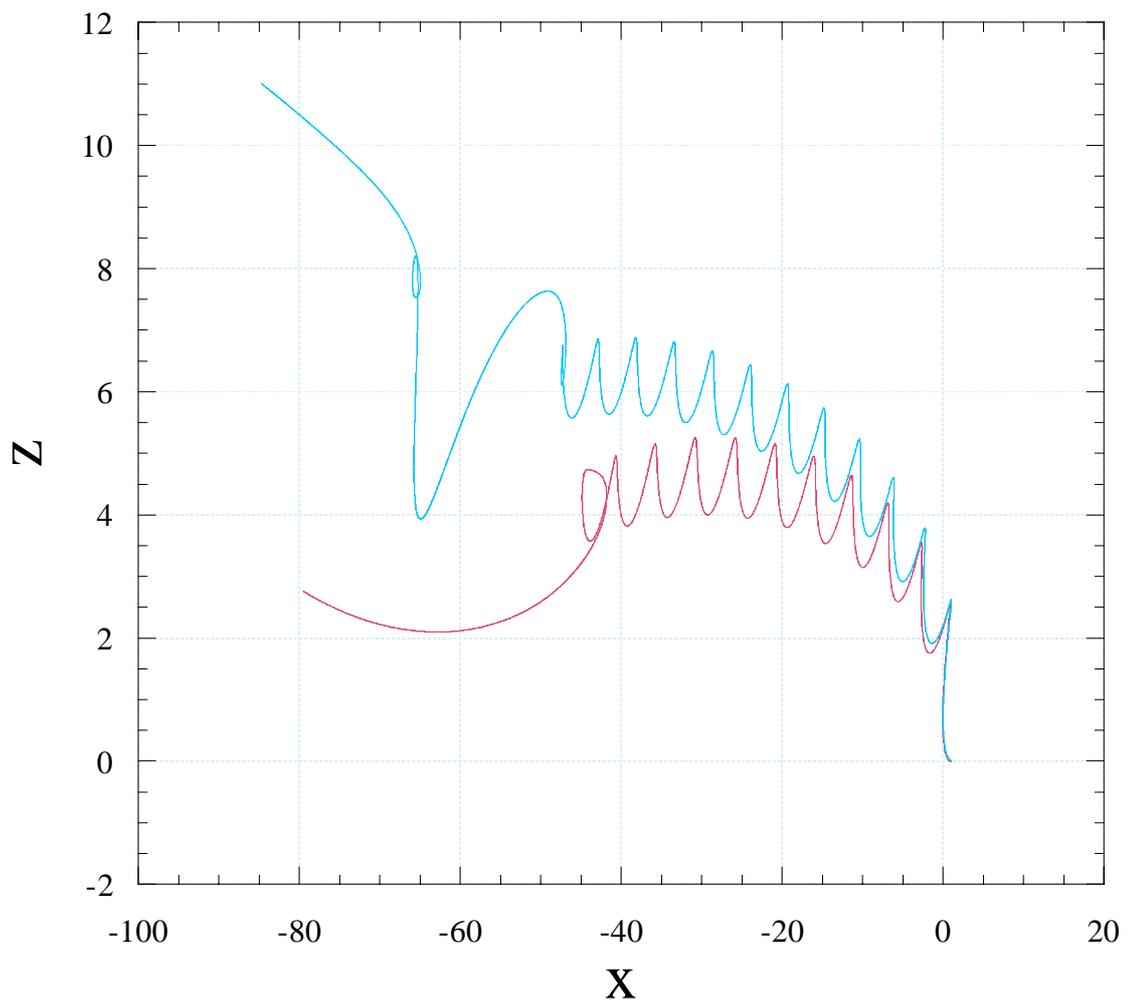


図 11: 1つ目の天体の  $x - z$  座標。  
 赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

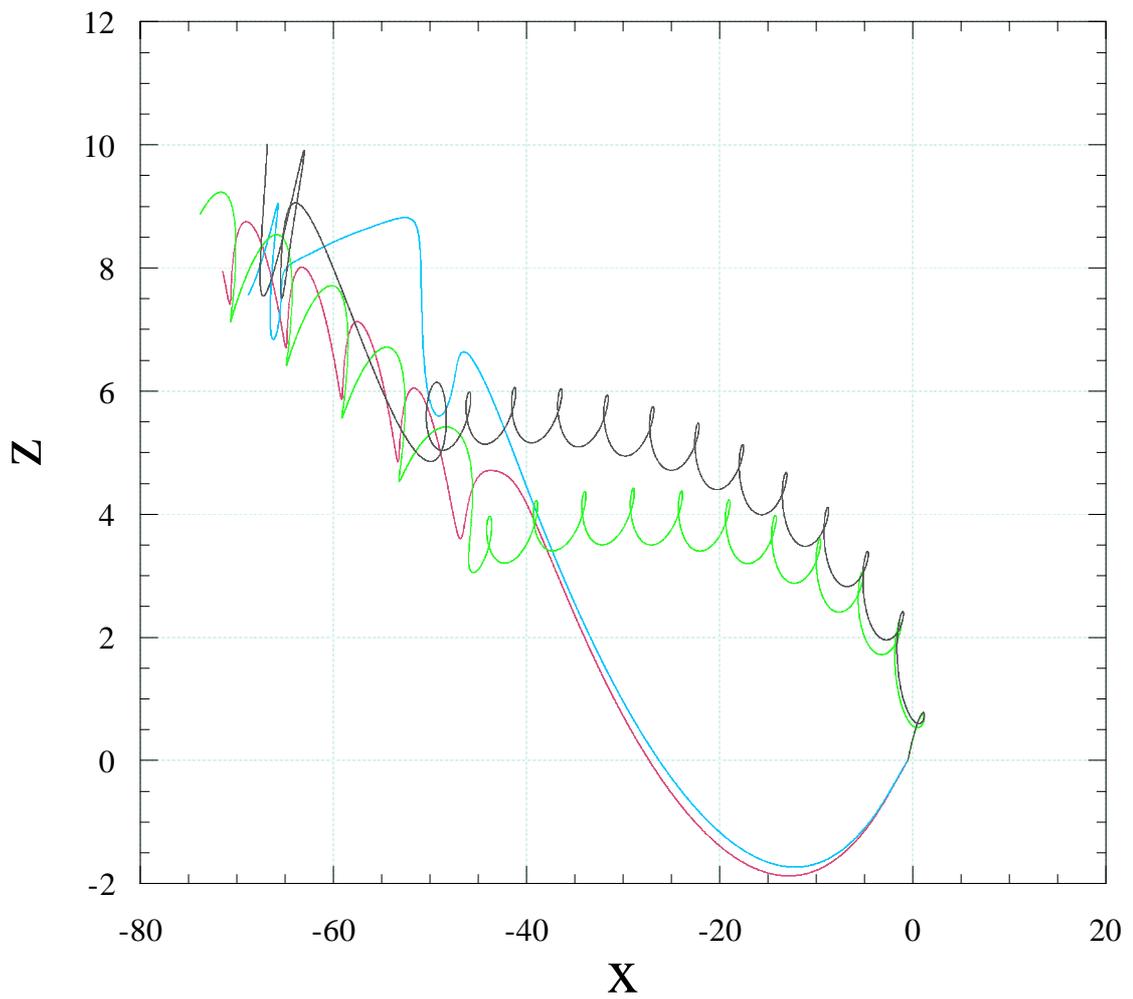


図 12: 2つ目と3つ目の天体の  $x - z$  座標。

赤:初めに与えた初期値での軌跡 (2つ目の天体), 青:初期値が  $\delta$  だけ異なる時の軌跡 (2つ目の天体),  
 緑:初めに与えた初期値での軌跡 (3つ目の天体), 黒:初期値が  $\delta$  だけ異なる時の軌跡 (3つ目の天体)。

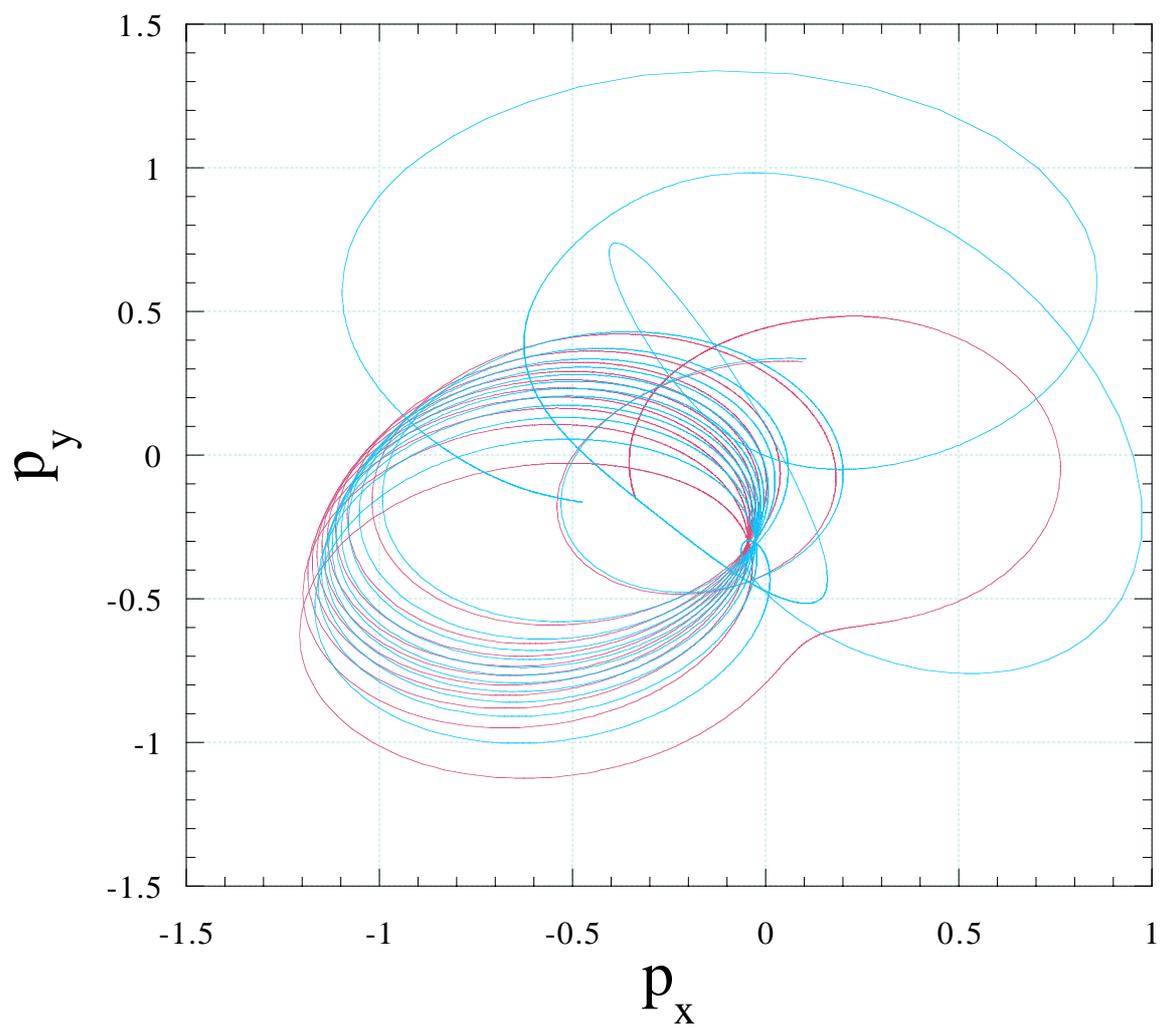


図 13: 1つ目の天体の  $p_x - p_y$  座標。  
赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

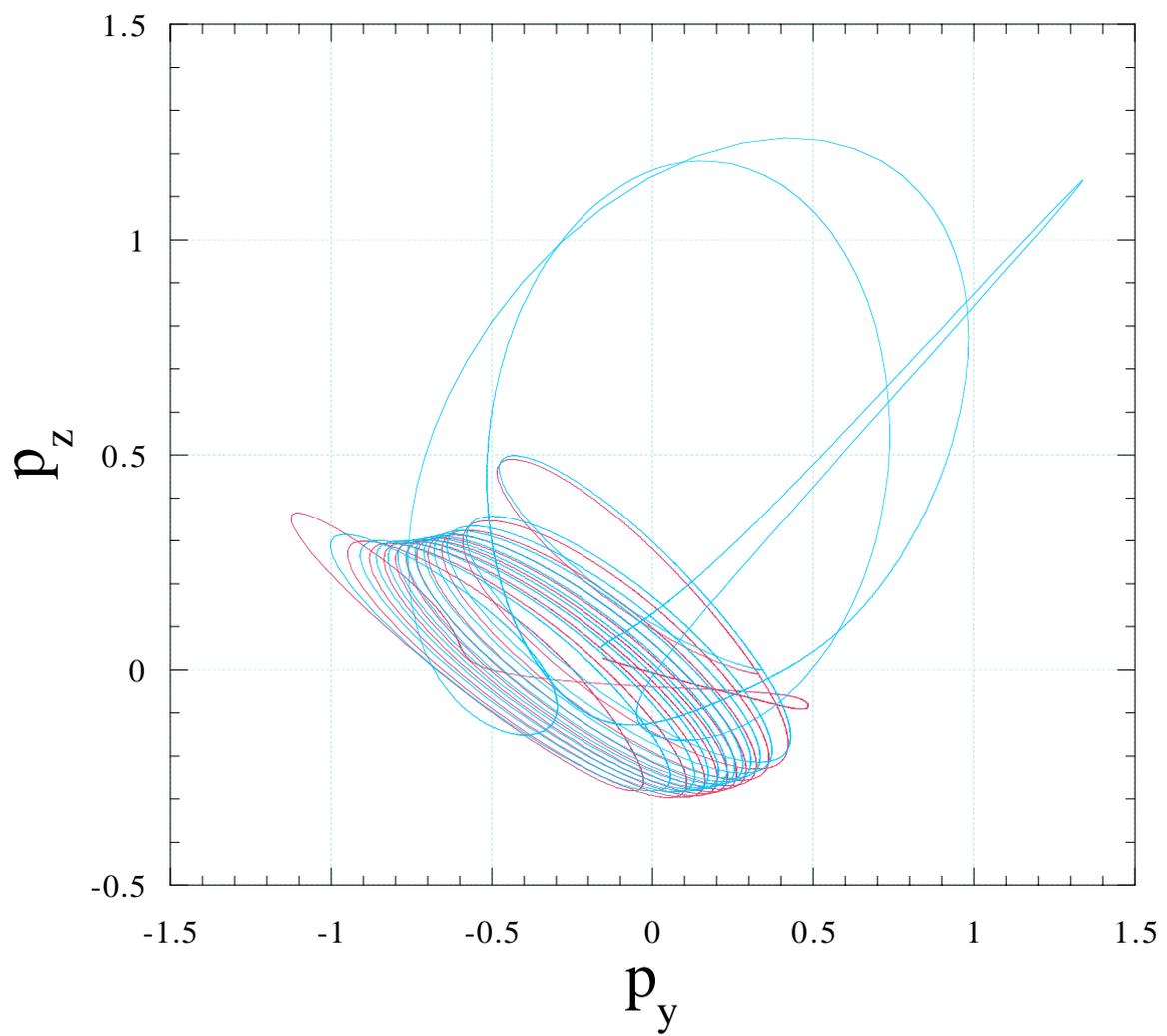


図 14: 1 つ目の天体の  $p_y - p_z$  座標。  
 赤: 初めに与えた初期値での軌跡, 青: 初期値が  $\delta$  だけ異なる時の軌跡。

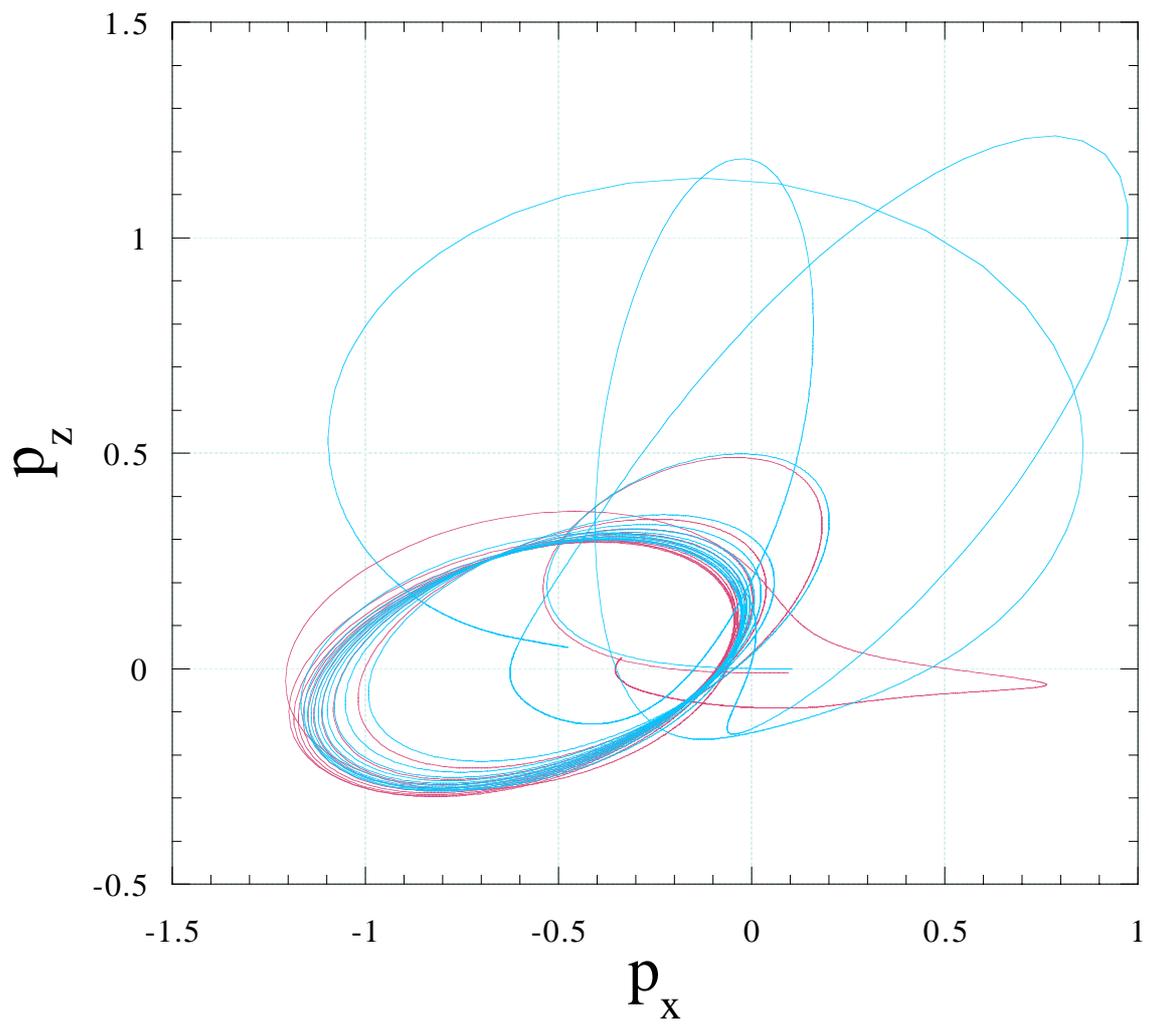


図 15: 1つ目の天体の  $p_x - p_z$  座標。  
 赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

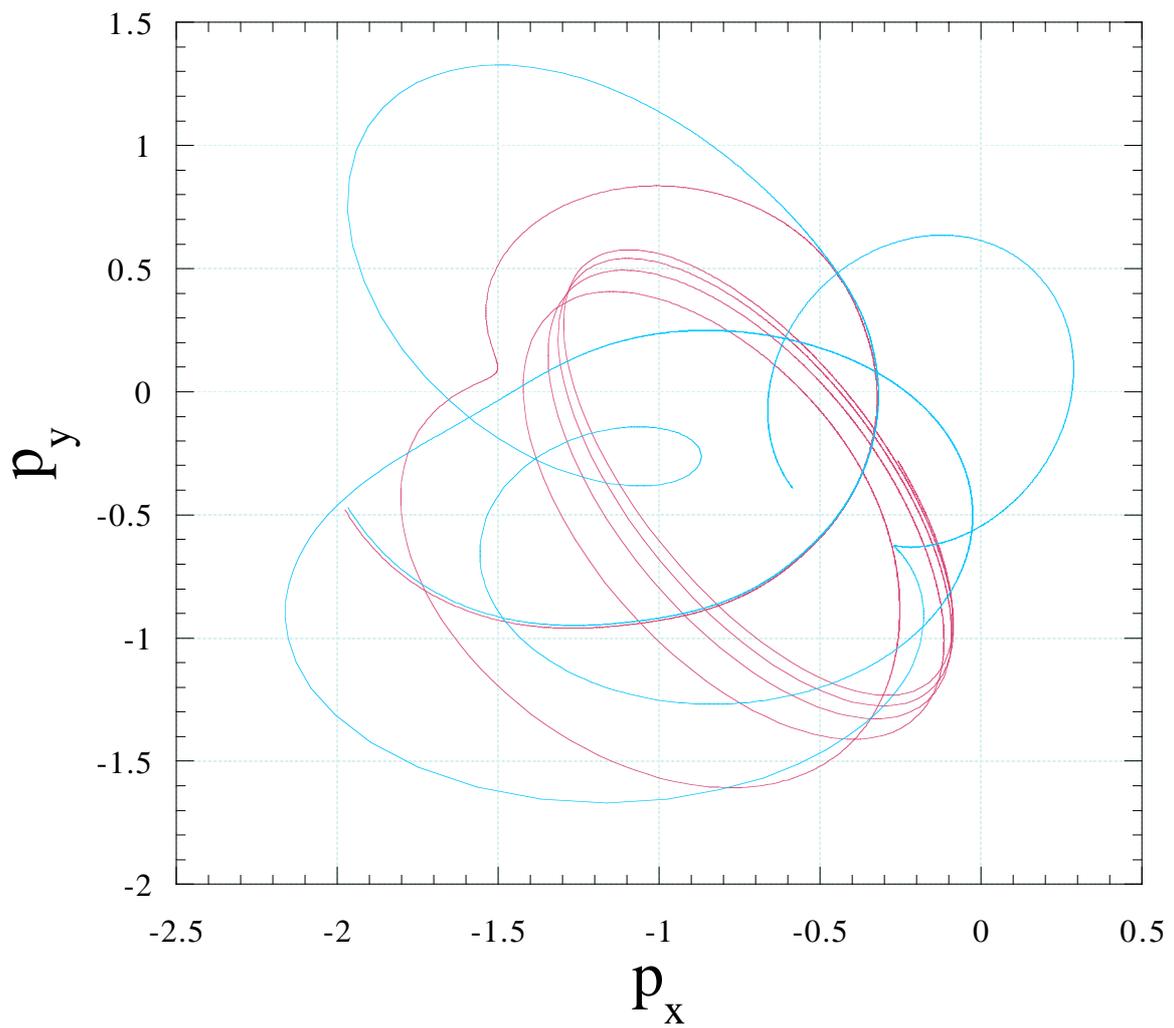


図 16: 2 つ目の天体の  $p_x - p_y$  座標。  
 赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

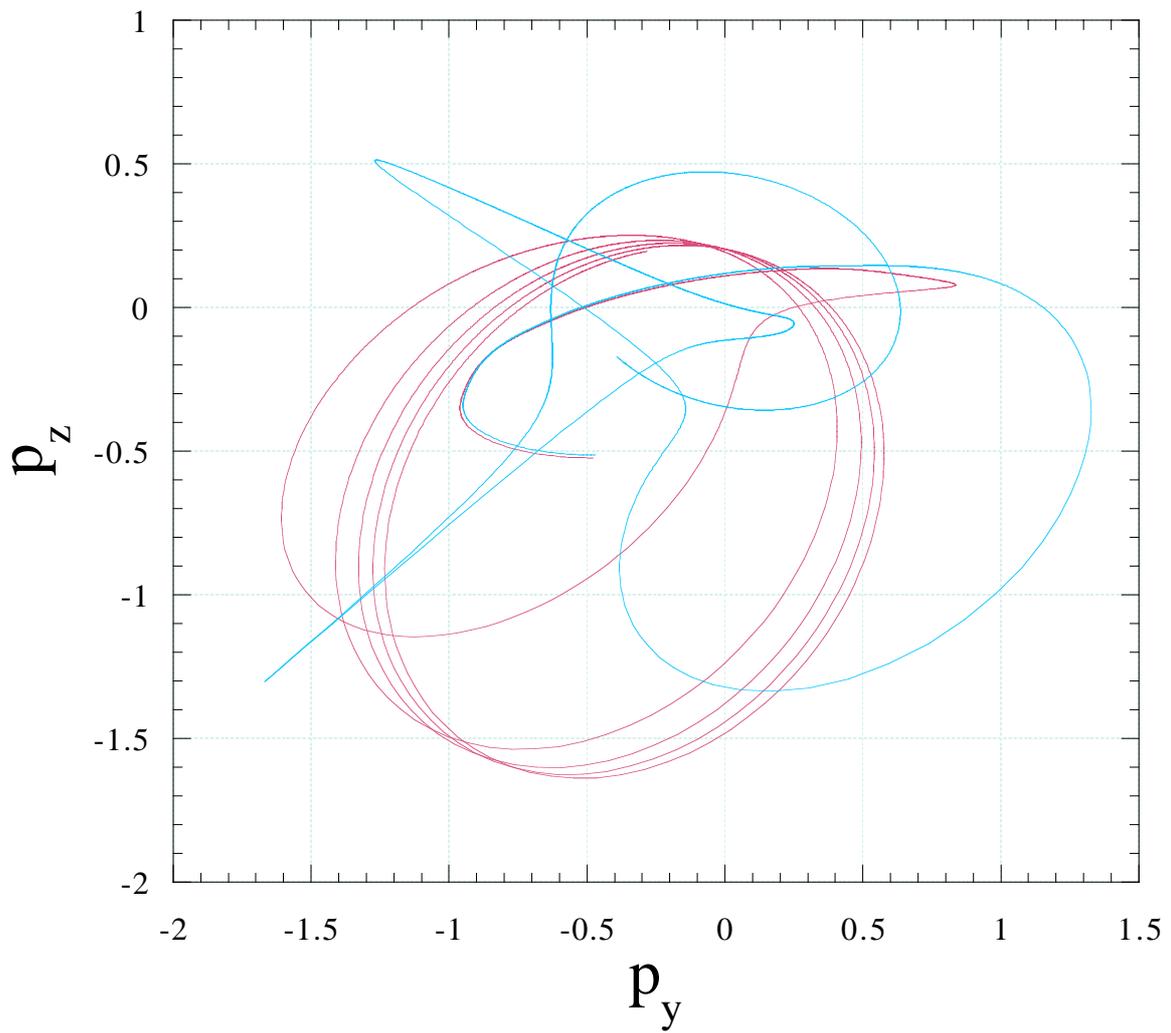


図 17: 2つ目の天体の  $p_y - p_z$  座標。  
 赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

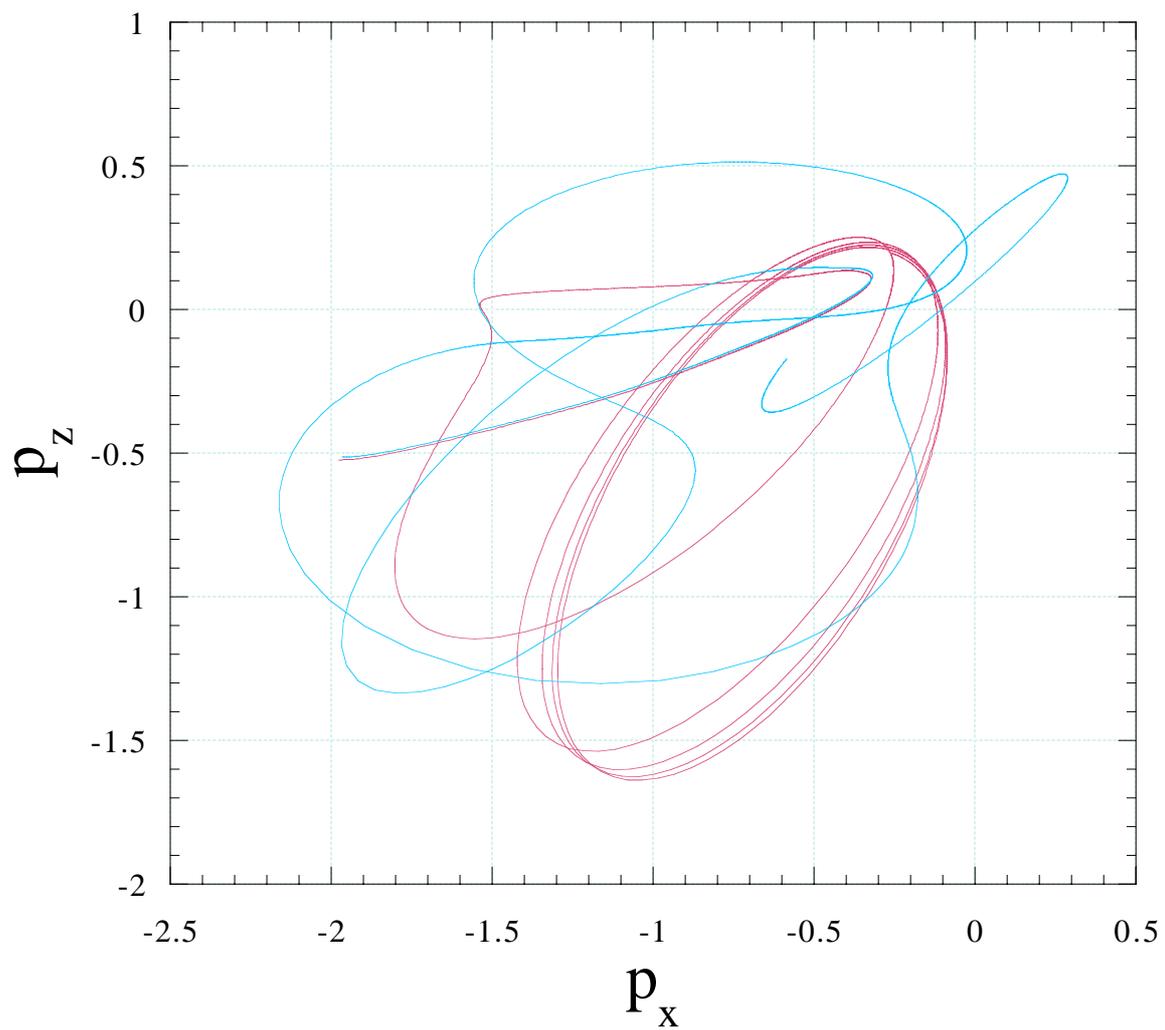


図 18: 2 つ目の天体の  $p_x - p_z$  座標。  
 赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

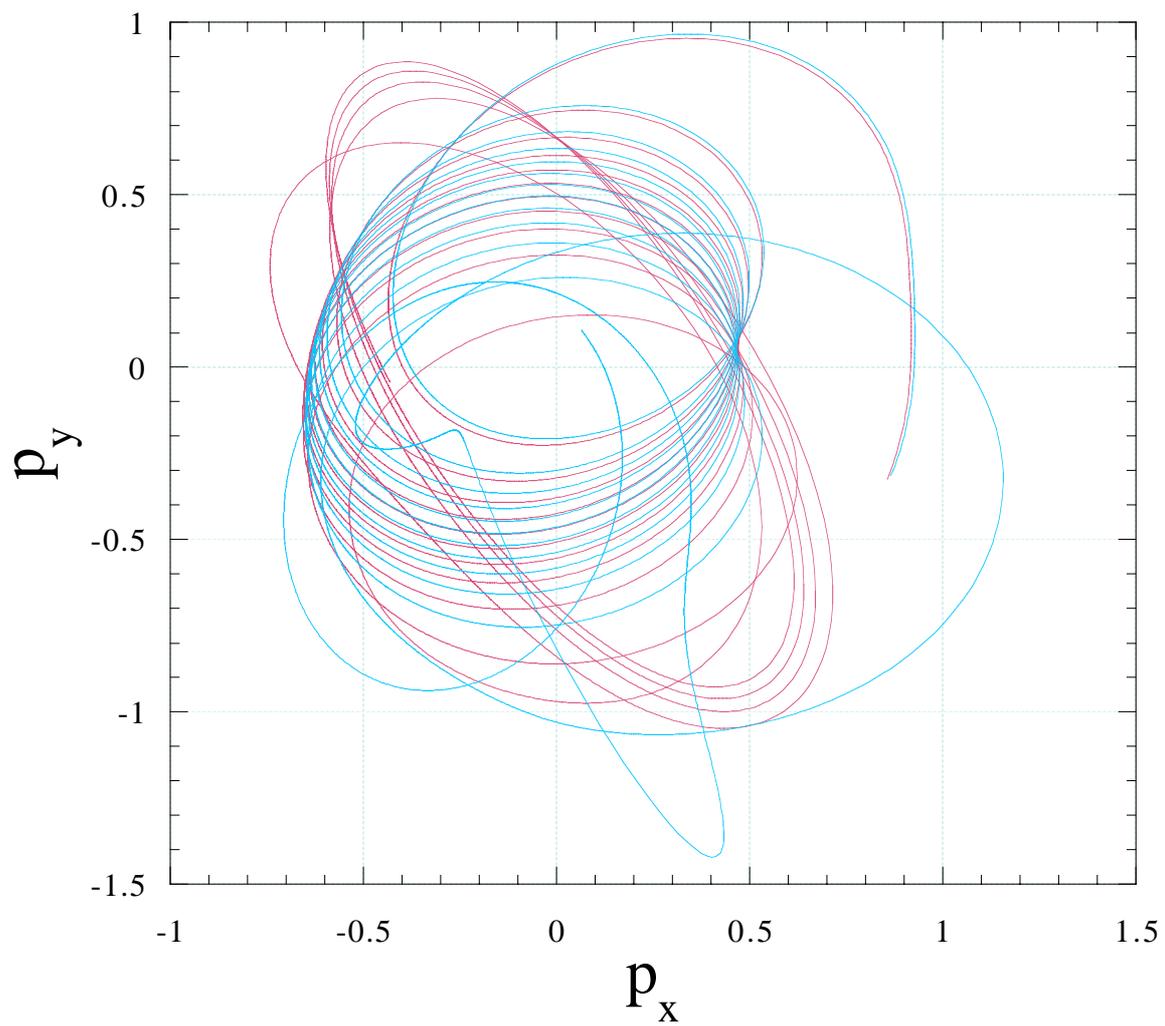


図 19: 3つ目の天体の  $p_x - p_y$  座標。  
赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

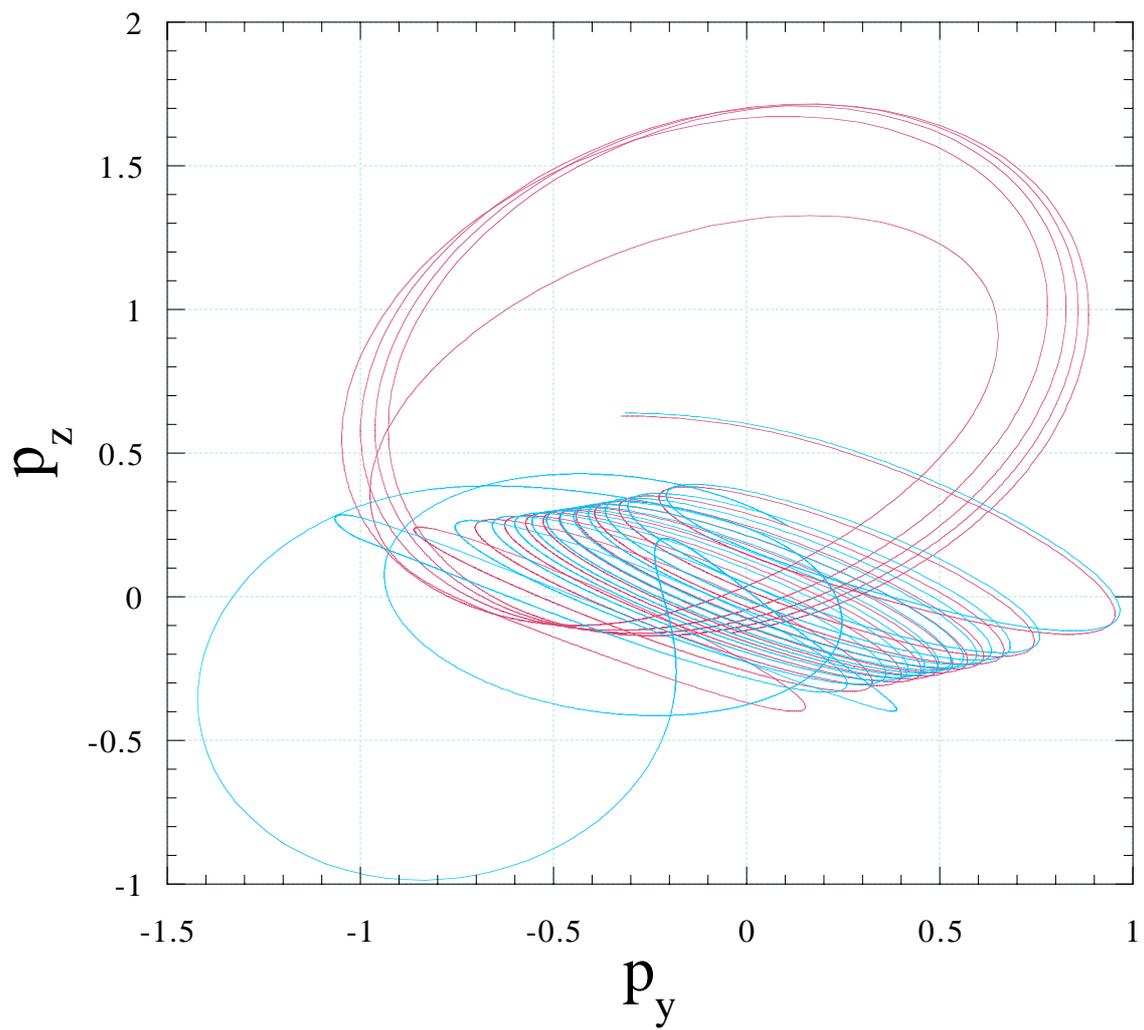


図 20: 3 つ目の天体の  $p_y - p_z$  座標。  
 赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

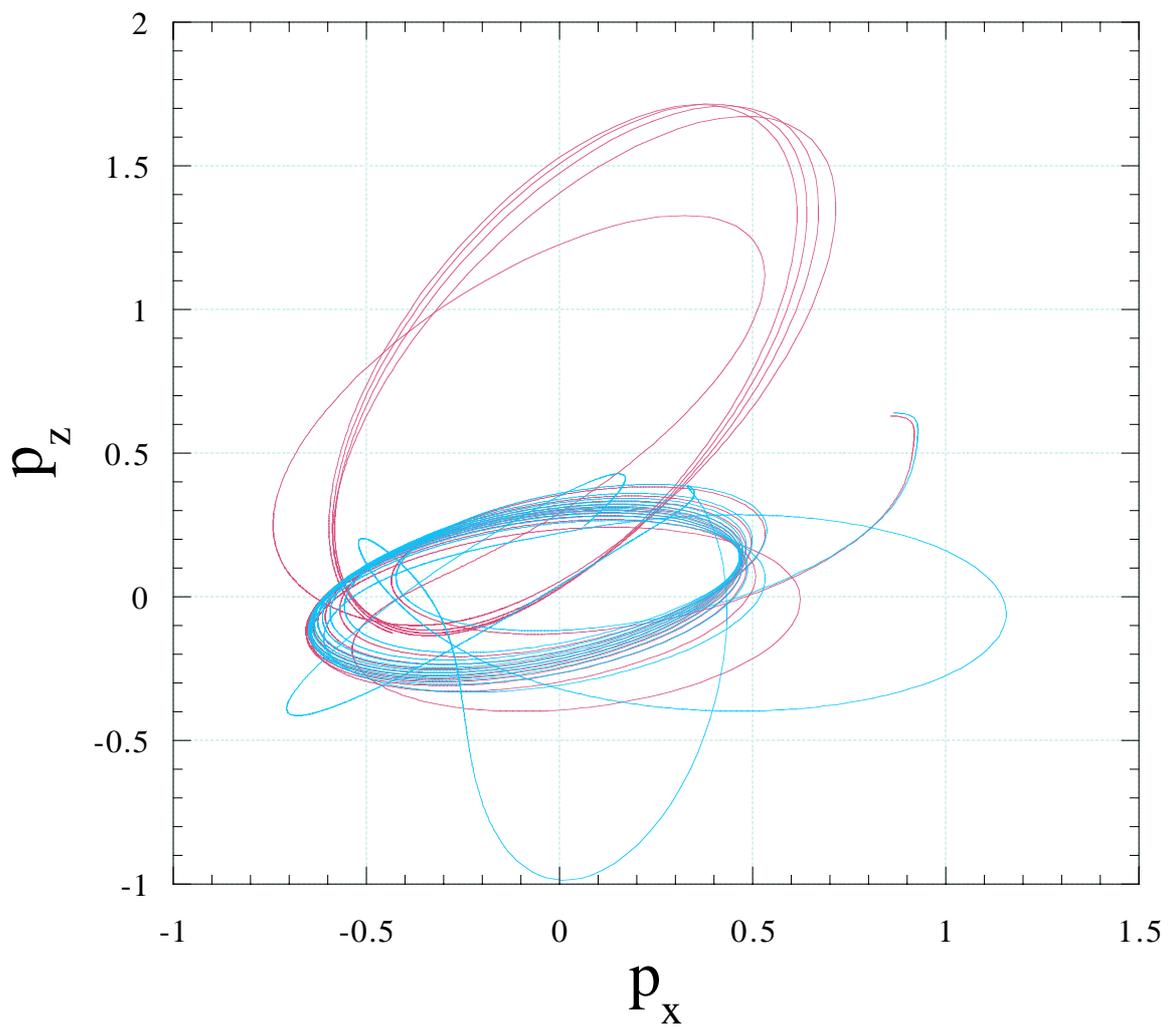


図 21: 3つ目の天体の  $p_x - p_z$  座標。  
 赤:初めに与えた初期値での軌跡, 青:初期値が  $\delta$  だけ異なる時の軌跡。

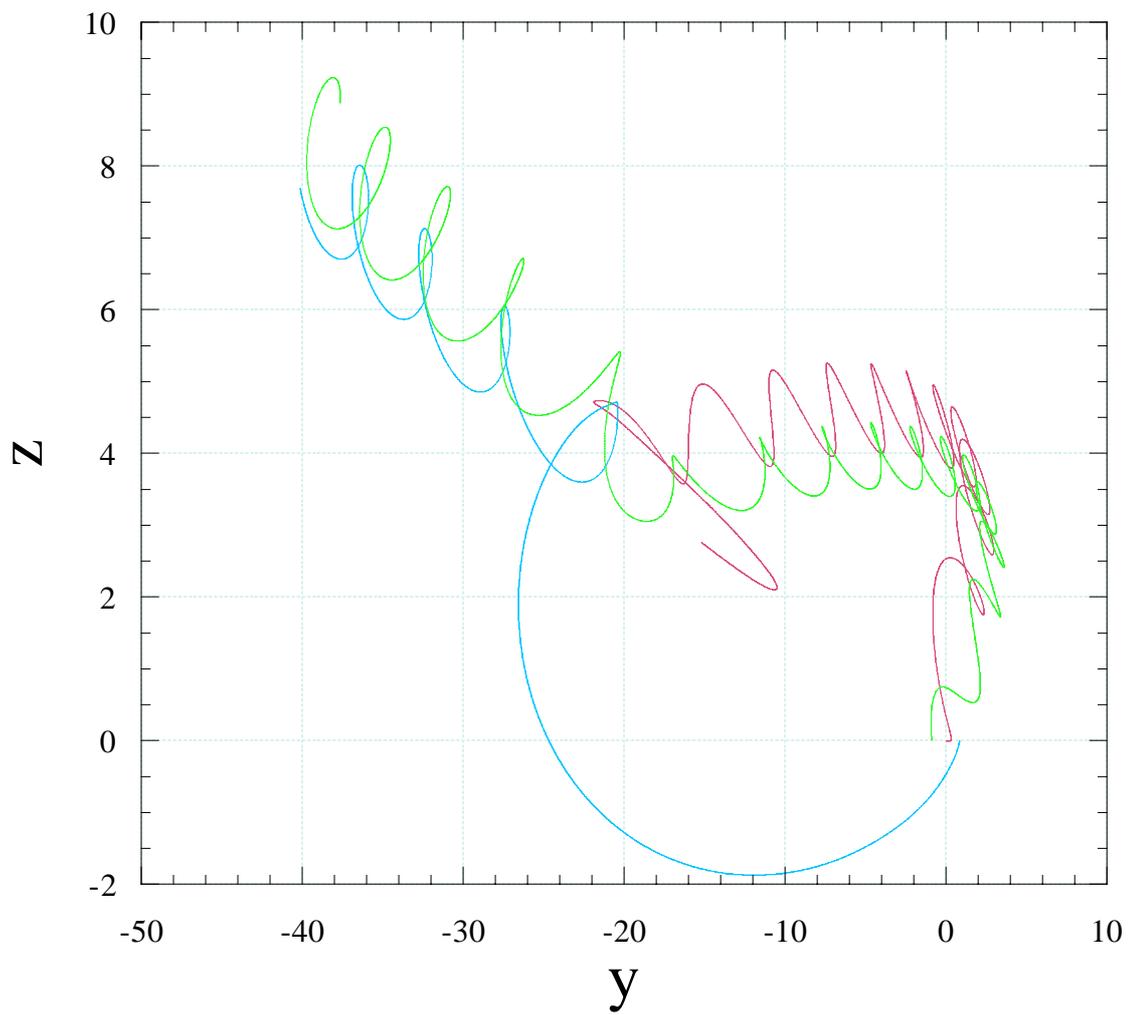


図 22: 3 天体の軌跡 ( $y - z$  座標)。  
 赤:1 つ目の天体, 青:2 つ目の天体, 緑:3 つ目の天体。

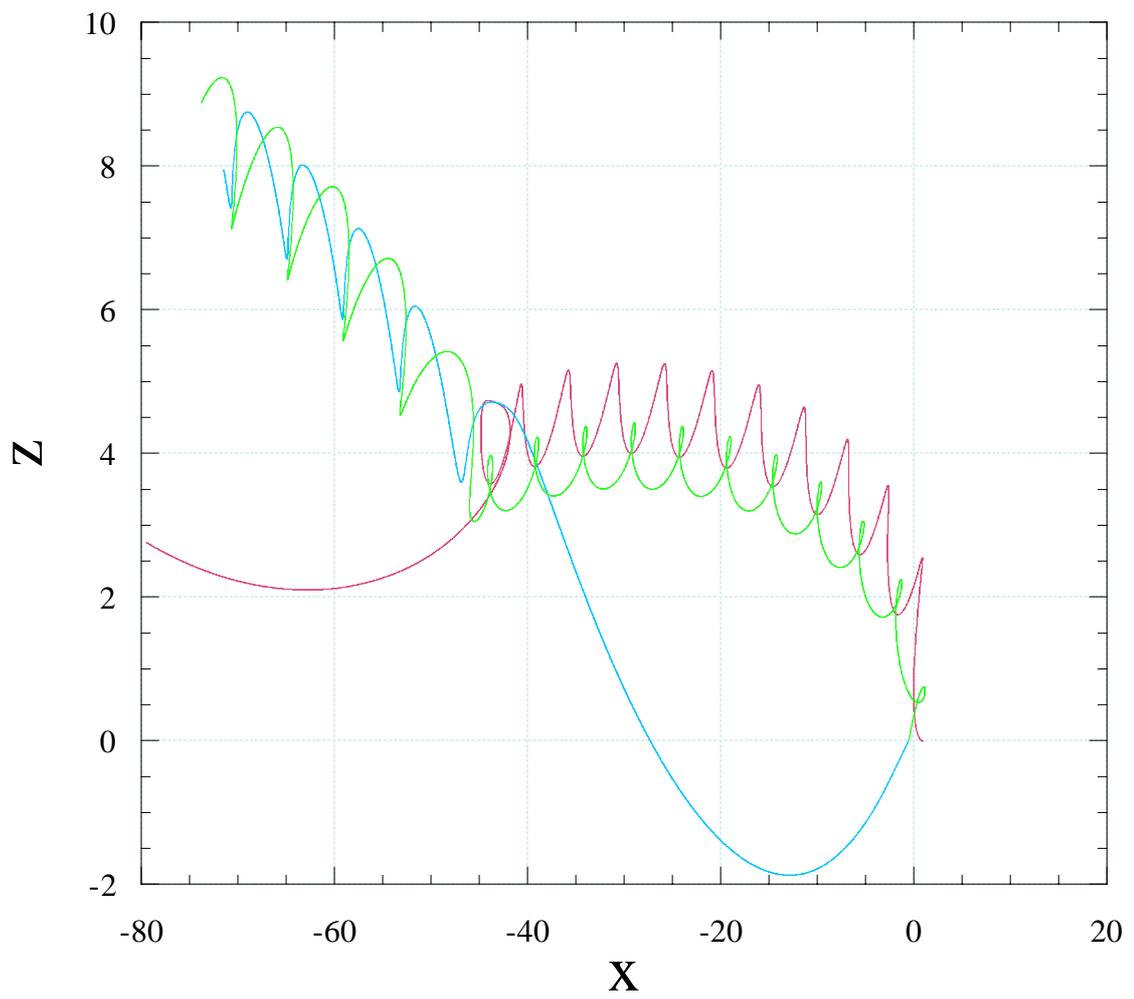


図 23: 3 天体の軌跡 ( $x - z$  座標)。  
 赤:1 つ目の天体, 青:2 つ目の天体, 緑:3 つ目の天体。

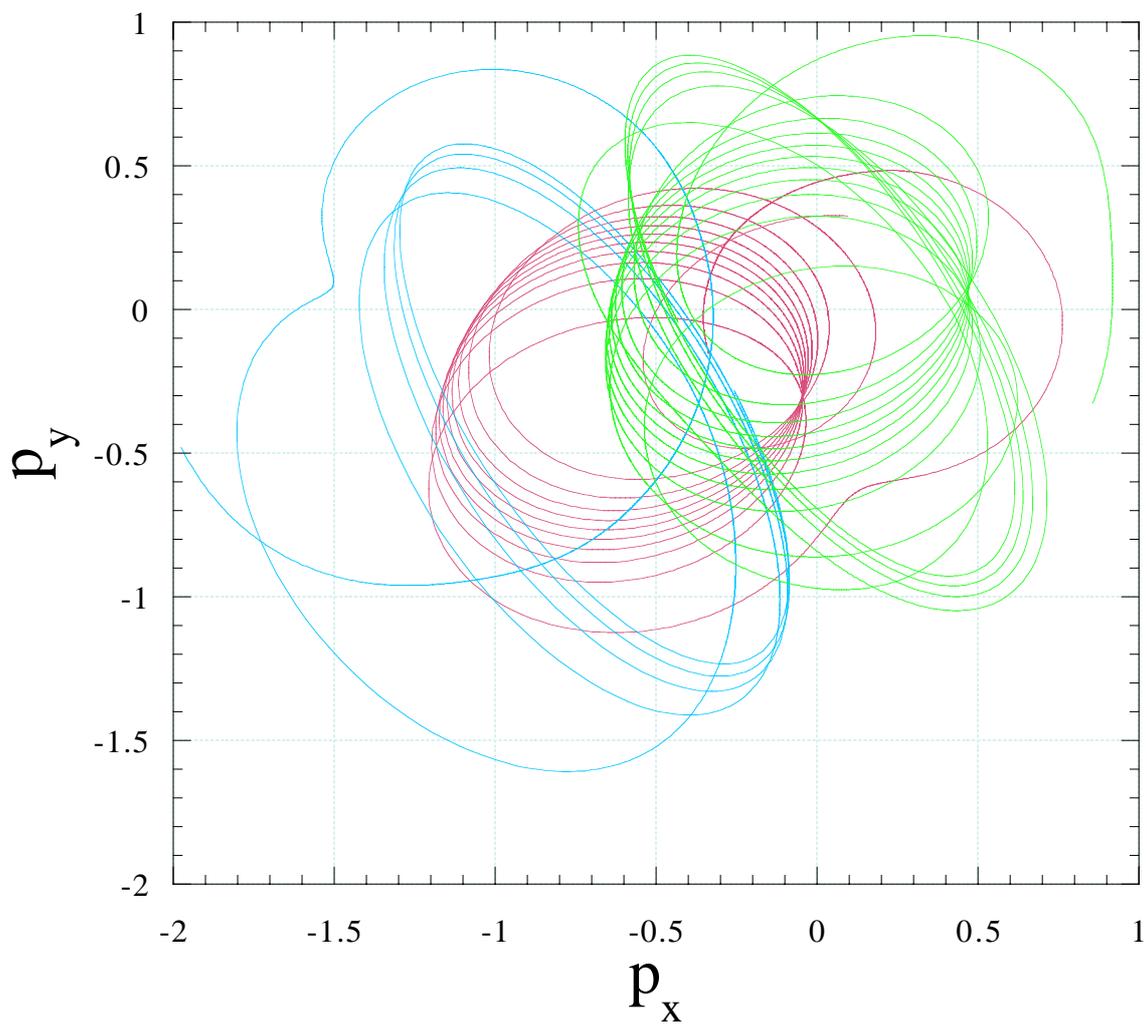


図 24: 3 天体の軌跡 ( $p_x - p_y$  座標)。  
赤:1 つ目の天体, 青:2 つ目の天体, 緑:3 つ目の天体。

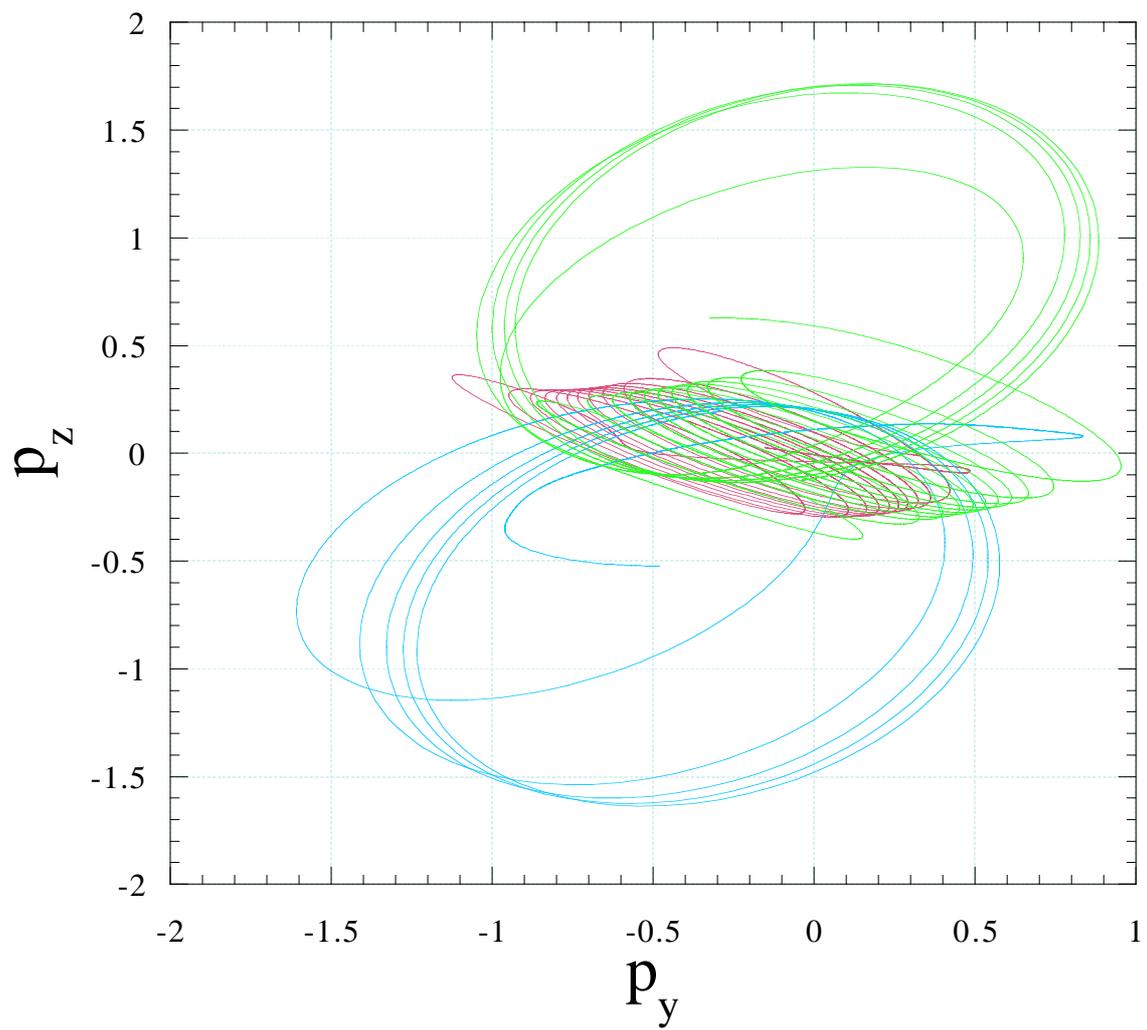


図 25: 3 天体の軌跡 ( $p_y - p_z$  座標)。  
赤:1 つ目の天体, 青:2 つ目の天体, 緑:3 つ目の天体。

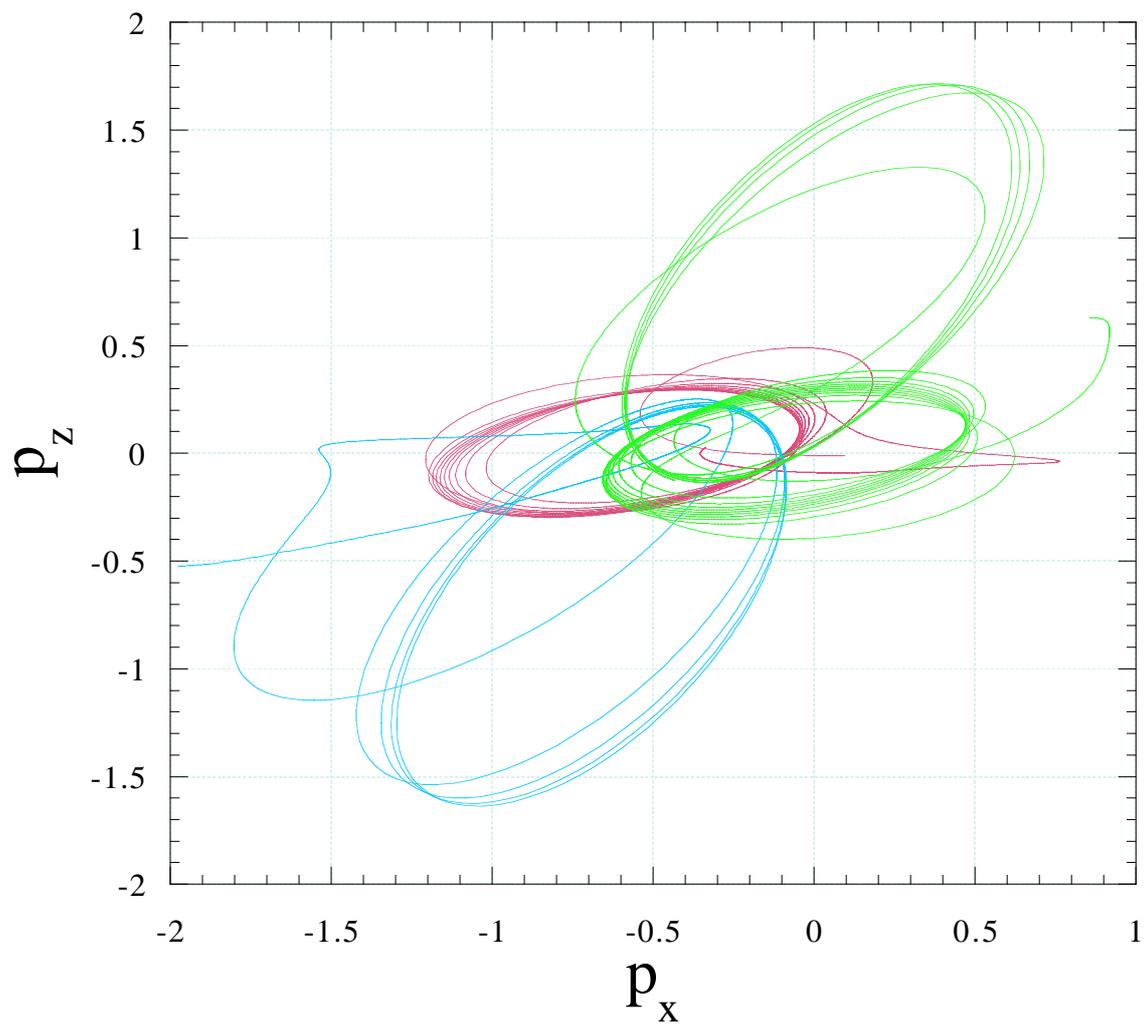


図 26: 3 天体の軌跡 ( $p_x - p_z$  座標)。  
赤:1 つ目の天体, 青:2 つ目の天体, 緑:3 つ目の天体。