

考虑带电粒子的电磁场对粒子本身反作用下 带电粒子在匀强磁场中低速运动 的数值模拟报告

欧纪阳*2019141220016 赵春丽 2019141220081

张凌凯 2019141220032 苗森 2019141220028 马卓瑜 2019141220045

College of Physics, Sichuan University, Chengdu 610064, China

2021 年 7 月 2 日

摘要

带电粒子与电磁场是相互作用的，一方面带电粒子激发电磁场，另一方面电磁场又对电荷有反作用，要完全解决带电粒子与电磁场系统的动力学问题，必须把两者直接的相互作用同时考虑。当考虑电带电粒子产生的辐射时，一部分能量和动量被电磁场带着，因此粒子的运动必然受到阻尼，在匀强磁场中的不再做圆周运动。本文将试图探究考虑带电粒子的电磁场对粒子本身的相互作用，即考虑辐射阻尼时粒子在匀强磁场中的运动，并在 Python 中使用龙格-库塔方法以及 Mathematica 进行数值模拟。

关键词：电动力学 电磁辐射 辐射阻尼 自身场 初值问题 Python Mathematica

1 介绍

在电磁学中，带电粒子在恒定电磁场中运动，会受到电场力与洛伦兹力的作用。当只存在匀强磁场时，根据带电粒子初速度的不同，粒子将做匀速圆周运动或者保持半径不变的螺旋运动，前者粒子在延磁场方向上不具有速度，后者则在该方向上具有速度。

而在学习了电动力学之后，应该明白在实际情况中，粒子的运动会辐射电磁场，一部分能量和动量被电磁场带着，因此粒子的运动必然受到阻尼。因此粒子的运动不是单纯被外场作用力决定的，粒子所激发的场对粒子本身也有作用力，为了完全解出粒子的运动，必须在粒子的动力学方程中除了包含除该粒子本身以外的场产生的电场力和洛伦兹力以外，还必须包含该粒子自身产生的辐射场的反作用力，即辐射阻尼力。

在这篇文章中，我们报告了计算带电粒子在包含自身辐射场的电磁场中低速运动的方法，并使用 Python 进行数值计算。在本文的第二部分中，将讨论计算辐射阻尼的方法；本文的第三部分将会介绍数值模拟的方法；不考虑辐射阻尼的数值模拟的结果将展示在本文的第四部分；我们最后在第五部分中利用 Mathematica 讨论存在辐射阻尼的情况。

本文的相关代码已上传至 GitHub 公开库<https://github.com/812610357/Electrodynamics>。

2 理论

在这一部分中将会补充一些电动力学知识，讨论从推迟势到电偶极辐射的功率，再到使用辐射功率推到辐射阻尼的方法，得到带电粒子在电磁场中的运动方程。

*E-mail: Julian-OU@outlook.com

2.1 电磁场的推迟势

把洛伦兹规范条件

$$\nabla \cdot \mathbf{A} + \frac{1}{c^2} \frac{\partial \varphi}{\partial t} = 0$$

以及用矢势 \mathbf{A} 、标势 φ 表示的磁场、电场表示式

$$\mathbf{B} = \nabla \times \mathbf{A} \quad \mathbf{E} = -\frac{\partial \mathbf{A}}{\partial t} - \nabla \varphi$$

带入麦克斯韦方程组中

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad \nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \quad \nabla \times \mathbf{B} = \mu_0 \varepsilon_0 \frac{\partial \mathbf{E}}{\partial t} + \mu_0 \mathbf{J} \quad \nabla \cdot \mathbf{B} = 0$$

可以得到达朗贝尔方程

$$\nabla^2 \mathbf{A} - \frac{1}{c^2} \frac{\partial^2 \mathbf{A}}{\partial t^2} = -\mu_0 \mathbf{J} \quad \nabla^2 \varphi - \frac{1}{c^2} \frac{\partial^2 \varphi}{\partial t^2} = -\frac{\rho}{\varepsilon_0}$$

分别求解后可以得到推迟势公式

$$\mathbf{A}(\mathbf{x}, t) = \frac{\mu_0}{4\pi} \int_V \frac{\mathbf{J}(\mathbf{x}', t - \frac{r}{c})}{r} dV' \quad \varphi(\mathbf{x}, t) = \frac{1}{4\pi\varepsilon_0} \int_V \frac{\rho(\mathbf{x}', t - \frac{r}{c})}{r} dV' \quad (1)$$

以上两式表明电磁作用具有一定的传播速度，空间某点 \mathbf{x} 在时刻 t 的场值不是依赖于同一时刻的电荷电流分布，而是决定于较早时刻

$$t' = t - \frac{r}{c}$$

的电荷电流分布。

2.2 低速运动粒子的辐射场

对带电粒子来说 $\mathbf{J} = \rho \mathbf{v}$ ，其中 \mathbf{v} 为粒子在辐射时刻 t' 的速度，由 (1) 式可知势依赖于粒子的速度 \mathbf{v} ，而不依赖于加速度 $\dot{\mathbf{v}}$ ，因此可以取一个在粒子辐射时刻相对静止的参考系 $\tilde{\Sigma}$ ，则在 $\tilde{\Sigma}$ 观察 $(\tilde{\mathbf{x}}, \tilde{t})$ 上势的瞬间值为

$$\tilde{\mathbf{A}} = 0 \quad \tilde{\varphi} = \frac{q}{4\pi\varepsilon_0 \tilde{r}}$$

其中 $\tilde{r} = c(\tilde{t} - \tilde{t}')$ 为 $\tilde{\Sigma}$ 上观察的粒子与场点的距离。运用洛伦兹变换把势变换回原参考系 Σ 中

$$\mathbf{A} = \frac{\mathbf{v} \tilde{\varphi}}{c^2 \sqrt{1 - \frac{v^2}{c^2}}} = \frac{1}{\sqrt{1 - \frac{v^2}{c^2}}} \frac{q \mathbf{v}}{4\pi\varepsilon_0 c^2 \tilde{r}} \quad \varphi = \frac{\tilde{\varphi}}{\sqrt{1 - \frac{v^2}{c^2}}} = \frac{1}{\sqrt{1 - \frac{v^2}{c^2}}} \frac{q}{4\pi\varepsilon_0 \tilde{r}}$$

其中 \tilde{r} 任然是 $\tilde{\Sigma}$ 上观察到的距离，用洛伦兹变换将其改写为 Σ 上的距离

$$\tilde{r} = (\tilde{t} - \tilde{t}') = \frac{c(t - t') - \frac{\mathbf{v}}{c} \cdot (\mathbf{x} - \mathbf{x}')}{\sqrt{1 - \frac{v^2}{c^2}}} = \frac{r - \frac{\mathbf{v}}{c} \cdot \mathbf{r}}{\sqrt{1 - \frac{v^2}{c^2}}}$$

因此得到势在 Σ 下的表达式

$$\mathbf{A} = \frac{q \mathbf{v}}{4\pi\varepsilon_0 c^2 \left(r - \frac{\mathbf{v}}{c} \cdot \mathbf{r} \right)} \quad \varphi = \frac{q}{4\pi\varepsilon_0 \left(r - \frac{\mathbf{v}}{c} \cdot \mathbf{r} \right)} \quad (2)$$

因为当低速运动 $v \ll c$ 时有

$$\frac{\partial t'}{\partial t} = \frac{r}{r - \frac{\mathbf{v} \cdot \mathbf{r}}{c}} = 1 \quad \nabla t' = -\frac{r}{c \left(r - \frac{\mathbf{v} \cdot \mathbf{r}}{c} \right)} = -\frac{\mathbf{r}}{cr} = -\frac{-\mathbf{e}_r}{c}$$

把势公式 (2) 式对时空坐标微分得

$$\begin{aligned} \mathbf{B} &= \nabla \times \mathbf{A} = \nabla \times \mathbf{A}|_{t'=const.} + \nabla t' \times \frac{\partial \mathbf{A}}{\partial t'} = \frac{q\mathbf{v} \times \mathbf{r}}{4\pi\epsilon_0 c^2 r^3} - \frac{\mathbf{r}}{cr} \times \frac{q\dot{\mathbf{v}}}{4\pi\epsilon_0 c^2 r} = \frac{q\mathbf{v} \times \mathbf{r}}{4\pi\epsilon_0 c^2 r^3} - \frac{q\dot{\mathbf{v}} \times \mathbf{r}}{4\pi\epsilon_0 c^3 r^2} \\ \mathbf{E} &= -\frac{\partial \mathbf{A}}{\partial t} - \nabla\varphi = -\frac{\partial t'}{\partial t} \frac{\partial \mathbf{A}}{\partial t'} - \nabla\varphi|_{t'=const.} - \nabla t' \frac{\partial \varphi}{\partial t'} = \frac{q\mathbf{r}}{4\pi\epsilon_0 r^3} - \frac{q\dot{\mathbf{v}}}{4\pi\epsilon_0 c^2 r} + \frac{\mathbf{r}}{cr} \frac{q\dot{\mathbf{v}} \cdot \mathbf{r}}{4\pi\epsilon_0 cr^2} = \frac{q\mathbf{r}}{4\pi\epsilon_0 r^3} + \frac{q\mathbf{r} \times (\mathbf{r} \times \dot{\mathbf{v}})}{4\pi\epsilon_0 c^2 r^3} \end{aligned}$$

其中与 r 成反比的项为辐射电磁场

$$\mathbf{B} = \frac{q\dot{\mathbf{v}} \times \mathbf{r}}{4\pi\epsilon_0 c^3 r^2} = \frac{q}{4\pi\epsilon_0 c^3} \dot{\mathbf{v}} \times \mathbf{e}_r \quad \mathbf{E} = \frac{q\dot{\mathbf{v}} \times \mathbf{r}}{4\pi\epsilon_0 c^3 r^2} = \frac{q}{4\pi\epsilon_0 c^2} \mathbf{e}_r \times (\mathbf{e}_r \times \dot{\mathbf{v}})$$

于是可以得到辐射能流密度以及辐射总功率

$$\mathbf{S} = \mathbf{E} \times \mathbf{B} = \frac{q^2 \dot{\mathbf{v}}^2}{16\pi^2 \epsilon_0 c^3 r^2} \sin^2 \Theta \mathbf{e}_r \quad P = \oint \mathbf{S} \cdot \mathbf{e}_r r^2 d\Omega = \frac{q^2 \dot{\mathbf{v}}^2}{6\pi\epsilon_0 c^2}$$

其中 Θ 表示 \mathbf{r} 与 $\dot{\mathbf{v}}$ 的夹角。

2.3 辐射阻尼力与运动方程

当带电粒子受外力作用二加速时, 粒子将辐射出电磁波, 把部分能量辐射出去, 因而粒子受到一个阻尼力, 设 \mathbf{F}_e 表示外力, \mathbf{F}_s 代表粒子激发的场对粒子本身的反作用力, 即辐射阻尼力, 则粒子的运动方程为

$$m\dot{\mathbf{v}} = \mathbf{F}_e + \mathbf{F}_s$$

其中辐射阻尼力对粒子所作的负功率应等于辐射功率

$$\mathbf{F}_s \cdot \mathbf{v} = -P = -\frac{q^2 \dot{\mathbf{v}}^2}{6\pi\epsilon_0 c^3} \quad (3)$$

(3) 式中, 粒子在某一瞬间时的速度 \mathbf{v} 与加速度 $\dot{\mathbf{v}}$ 一般是不相关的量, (3) 式右边不能表示为一个力乘上速度 \mathbf{v} 的形式, 因此 (3) 式不可能对每一瞬间成立, 但是在粒子作准圆周运动时, 当粒子运动一周后, 粒子附近的场回到原来的状态, 可以考虑使用平均阻尼效应, 即 (3) 式对一周积分是成立的

$$\int_{t_0}^{t_0+T} \mathbf{F}_s \cdot \mathbf{v} dt = \int_{t_0}^{t_0+T} -\frac{q^2 \dot{\mathbf{v}}^2}{6\pi\epsilon_0 c^3} dt = -\frac{q^2}{6\pi\epsilon_0 c^3} \dot{\mathbf{v}} \cdot \mathbf{v} \Big|_{t_0}^{t_0+T} + \int_{t_0}^{t_0+T} \frac{q^2 \ddot{\mathbf{v}} \cdot \mathbf{v}}{6\pi\epsilon_0 c^3} dt$$

其中第一项为零, 因为当粒子运动一周后, $\dot{\mathbf{v}}$ 与 \mathbf{v} 回到原值, 因此辐射阻尼力可取

$$\mathbf{F}_s = \frac{q^2 \ddot{\mathbf{v}}}{6\pi\epsilon_0 c^3}$$

于是便得到了带电粒子在匀强磁场中的运动方程

$$m\dot{\mathbf{v}} = q\mathbf{v} \times \mathbf{B} + \frac{q^2 \ddot{\mathbf{v}}}{6\pi\epsilon_0 c^3} \quad (4)$$

3 方法

由于题目所给的带电粒子质量和电荷的数量级导致辐射阻尼力的数量级为 10^{-14} 次方, 小于机器精度, 因此在使用 Python 编写时将不考虑辐射阻尼力, 但是在报告的第四部分中, 将给出使用 Mathematica 求解的结果对比。

Input: f : 导数对应的函数
 t : 求解范围
 h : 求解步长
 w_0 : $t = 0$ 时的初值

```

1  $n = \text{int}(t/h)$ 
2  $t_0 = 0$ 
3 for  $i = 0, 1, 2, \dots, n-1$  do
4    $s1 = f(t_i, w_i)$ 
5    $s2 = f(t_i + h/2, w_i + s1 * h/2)$ 
6    $s3 = f(t_i + h/2, w_i + s2 * h/2)$ 
7    $s4 = f(t_i + h, w_i + s3 * h)$ 
8    $w_{i+1} = w_i + (s1 + 2 * s2 + 2 * s3 + s4) * h/6$ 
9    $t_{i+1} = t_i + h$ 
10 end

```

Output: t_0, t_1, \dots, t_n : 自变量序列
 w_0, w_1, \dots, w_n : 方程组的解

Algorithm 1: 4 阶龙格库塔方法

在上一部分中, 通过推导得到了带电粒子的运动方程 (4) 式去掉辐射阻尼力后为一个二阶矢量偏微分方程, 可以改写为三个标量方程

$$m\ddot{\mathbf{r}} = q\dot{\mathbf{r}} \times \mathbf{B} \Rightarrow \begin{cases} m\ddot{x} = \dot{y}B_z - \dot{z}B_y \\ m\ddot{y} = \dot{z}B_x - \dot{x}B_z \\ m\ddot{z} = \dot{x}B_y - \dot{y}B_x \end{cases}$$

通过重新定义变量, 使上述三个二阶常微分方程变为六个一阶常微分方程

$$\begin{cases} w_{x1} = \dot{x}, w_{x2} = \ddot{x} \\ w_{y1} = \dot{y}, w_{y2} = \ddot{y} \\ w_{z1} = \dot{z}, w_{z2} = \ddot{z} \end{cases} \Rightarrow \begin{cases} w'_{x1} = w_{x2}, w'_{x2} = w_{y2}B_z - w_{z2}B_y \\ w'_{y1} = w_{y2}, w'_{y2} = w_{z2}B_x - w_{x2}B_z \\ w'_{z1} = w_{z2}, w'_{z2} = w_{x2}B_y - w_{y2}B_x \end{cases} \quad (5)$$

对以上六个方程组成的方程组, 应用 4 阶龙格-库塔方法进行求解, 具体算法如 Algorithm1 所示。

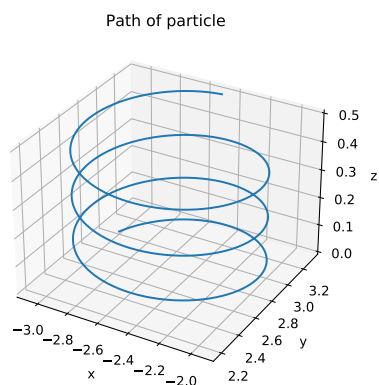
4 结果

利用 Python 编写 4 阶龙格-库塔方法, 代入 (5) 式中的方程组, 并设置初始条件为

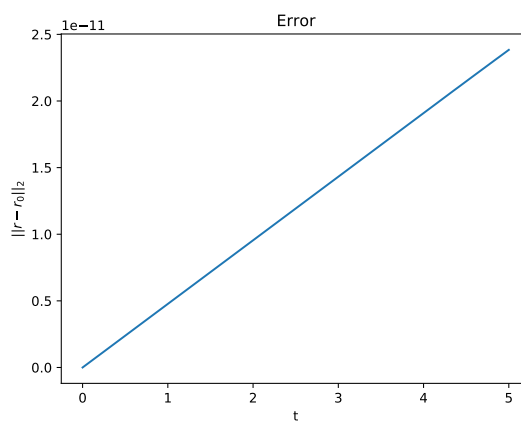
$$\begin{cases} w_{x1} = -3, w_{x2} = 1 \\ w_{y1} = 3, w_{y2} = 2 \\ w_{z1} = 0, w_{z2} = 0.1 \end{cases}$$

时间设置为 $t = 5$, 步长设置为 $h = 0.001$ 进行求解, 粒子的运动轨迹如图1(a) 所示。该问题具有解析解, 按以上初始条件下的解析解如 (6) 式, 计算二范数误差绘制, 如图1(b) 所示, 计算得到的粒子距离原点的最小距离约为 3.15872071, 解析解所给出的最小距离为 3.15871956。

$$\begin{cases} x = \frac{1}{4}(-2 \cos 4t + \sin 4t - 10) \\ y = \frac{1}{4}(\cos 4t + 2 \sin 4t + 11) \\ z = 0.1t \end{cases} \quad (6)$$



(a) 粒子的运动轨迹



(b) 误差曲线

图 1: 数值计算结果

5 讨论

由于数量级差别巨大, 无法在 Python 中利用 4 阶龙格-库塔方法进行求解, 在此利用商业数学软件 Mathematica 求解微分方程组进行求解, 设置相同的初始条件, 结果如图2所示。计算结果表明, 阻尼力的存在使粒子的运动轨迹不再以圆轨道螺旋上升, 而是做半径逐渐减小的螺旋上升运动, 这是由于粒子运动辐射出电磁波, 使其能量和动量损失所造成的。但是由于辐射功率较低, 导致辐射阻尼力远小于洛伦兹力, 只有在粒子长时间运动后其运动的半径才会有明显的变化。

6 鸣谢

感谢以下基金对本工作的支持, 国家自然科学基金: 11875227, 12047573, 11875196; 波谱与原子分子物理国家重点实验室开放基金: T151603

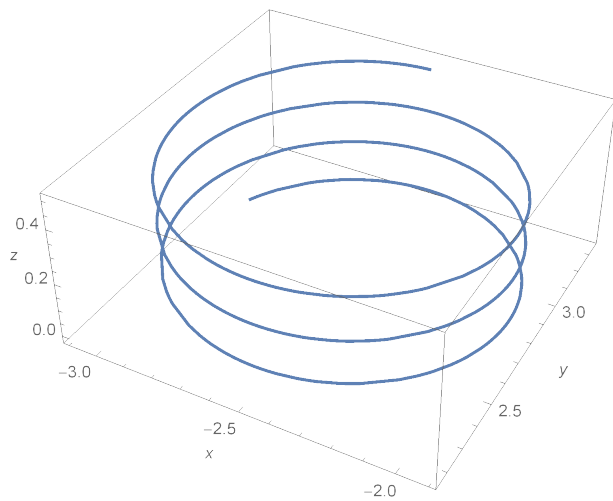
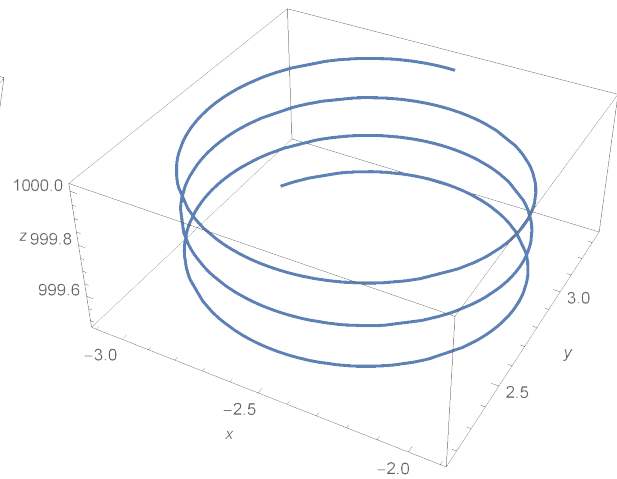
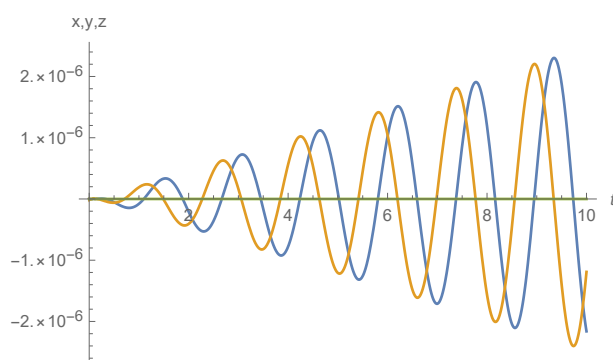
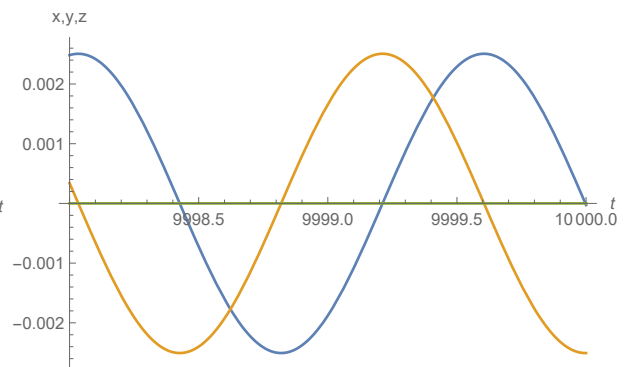
参考文献

- [1] 郭硕泓, 电动力学 (第三版), 高等教育出版社, 2008. 6
- [2] Sauer, T., 数值分析 (原书第二版), 工业出版社, 2014. 10

附代码

不考虑辐射阻尼的 Python 代码如下, 原始代码文件以及 Mathematica 笔记本在附文件中可以找到。

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 m = 1
5 q = 1
6 B = np.array([0, 0, 4])
7 r0 = np.array([-3, 3, 0]) # 初始位置
8 v10 = np.array([1, 2, 0.1]) # 初始速度
9 v20 = q*np.cross(v10, B)/m # 初始加速度
```

(a) $t = 0 \sim 5$ 时的粒子运动轨迹(b) $t = 9995 \sim 10000$ 时的粒子运动轨迹(c) $t = 0 \sim 10$ 时的计算结果对比图(d) $t = 9998 \sim 10000$ 时的计算结果对比图

(a) 和 (b) 为不同时间的段的数值计算结果；(c) 和 (d) 展示了不同时间段内，考虑了辐射阻尼与未考虑辐射阻尼的计算结果差异（用考虑了阻尼的计算结果减去未考虑的），蓝色线条表示 x 方向，橙色线条表示 y 方向，绿色线条表示 z 方向。

图 2: Mathematica 的求解结果

```

10 y0 = np.append(r0, v10)
11
12 c = 2.99792458e8
13 epsilon = 8.854187817e-12
14 k = q**2/(6*np.pi*epsilon*c**3)
15
16
17 def f(t, x):
18     x1 = x[:3]
19     x2 = x[3:]
20     s1 = x2
21     s2 = q*np.cross(x2, B)/m
22     s = np.append(s1, s2)
23     return s
24
25
26 def RK4(f, t, h, y0):
27     n = int(t/h)
28     t = np.linspace(0, t, n+1)
29     w = np.zeros((len(y0), n+1))

```

```
30     w[:, 0] = np.array([y0])
31     for i in range(n):
32         s1 = f(t[i], w[:, i])
33         s2 = f(t[i] + h/2, w[:, i] + s1*h/2)
34         s3 = f(t[i] + h/2, w[:, i] + s2*h/2)
35         s4 = f(t[i] + h, w[:, i] + s3*h)
36         w[:, i+1] = w[:, i] + (s1 + 2*s2 + 2*s3 + s4)*h/6
37     return t, w
38
39
40 t, r = RK4(f, 5, 1e-3, y0)
41 fig = plt.figure()
42 ax = fig.add_subplot(111, projection='3d')
43 ax.plot(r[0], r[1], r[2])
44 ax.set_xlabel('x')
45 ax.set_ylabel('y')
46 ax.set_zlabel('z')
47 ax.set_title('Path of particle')
48 plt.savefig('p.pdf')
49 plt.show()
50
51
52 def g(t):
53     x = (-10 - 2*np.cos(4*t) + np.sin(4*t))/4
54     y = (11 + np.cos(4*t) + 2*np.sin(4*t))/4
55     z = 0.1*t
56     return np.array([x, y, z])
57
58
59 r0 = g(t)
60 error = np.linalg.norm(r[:3]-r0, ord=2, axis=0)
61 plt.plot(t, error)
62 plt.title('Error')
63 plt.xlabel('t')
64 plt.ylabel('$||r-r_0||_2$')
65 plt.savefig('e.pdf')
66 plt.show()
67
68 print(np.min(np.linalg.norm(r[:3], ord=2, axis=0)))
```
