Devoir inspiré d'un article de Pierre Boissel dans le *Bulletin de l'Union des Physiciens numéro 741*. Nous nous interessons ici à l'aspect théorique et aux simulations numériques.

0 - Introduction

Nous nous proposons d'étudier le mouvement d'une bille qui rebondit sur un plateau vibreur. Ce système est modélisé par une bille ponctuelle de masse m, assujetie à un mouvement undimensionnel le long de l'axe vertical (noté z). Le mouvement de la bille n'influe pas sur celui du plateau. La bille n'est soumise qu'à deux forces : son poids et la force exercée par le vibreur lors des chocs. L'accélération de la pesanteur sera prise égale à $g = 10\ m.s^{-2}$. Le mouvement vertical du vibreur est sinusoïdal de pulsation Ω et d'amplitude $A > 0$. La position du plateau est donc toujours donnée par $\mathcal{A}(t) = \mathrm{A}\cos(\Omega t)$.

Ce système simple en apparence est le support de phénomènes dynamiques extrèmement riches. En particulier, nous pourrons y découvrir une transition vers le chaos par doublements successifs de période (voir figure 1a). Ainsi qu'une dynamique éventuellement gouvernée (selon les paramètres de contrôle) par des attracteurs étranges qui sont des objets physico-mathématiques toujours étudiés en recherche aujourd'hui (voir figure 1b).

L'expérience correspondante existe à l'Université de Jussieu et a été dupliquée au DER de physique de l'ENS Paris-Saclay. Une vidéo montrant les différentes dynamiques possibles de la bille est consultable ici : https://www.youtube.com/watch?v=4R_8_hQ2ihM

Une bille métallique d'une masse de 1g environ, maintenue latéralement par une languette souple, rebondit sur une céramique piezoélectrique attachée sur un vibreur. La cale piezo convertit les chocs en impulsions éléctriques ce qui permet de mesure les instants et les intensités des chocs. Certaines paramétrisations du modèle présenté ci-dessous ont été choisies précisément pour coller aux données issues de cette l'expérience.

1 - Description

Dans ce devoir, nous supposerons que le choc sur le vibreur est instantané, c'est à dire que la bille , lorsqu'elle rebondit, ne reste pas collée, même un court instant. Les différentes variables sont indicées par n, le numéro du choc. Au contact du vibreur, la bille arrivée au temps t_n avec une vitesse finale v_n^f, rebondit avec un coefficient de restitution noté μ ce qui lui donne une nouvelle vitesse initiale v_n^i telle que

$$v_n^i - \dot{\mathcal{A}}(t_n) = -\mu\left[v_n^f - \dot{\mathcal{A}}(t_n)\right]$$

où $\dot{\mathcal{A}}$ est la dérivée temporelle de l'amplitude, soit la vitesse du vibreur.

1. Justifier $0 \leq \mu \leq 1$. A quoi correspondent les cas limites ? Par la suite, on prendra μ=0.53.

Reponse: On a $\mu \geq 0$ car la bille doit changer de direction, avec $\mu = 0$ correspondant au cas où toute l'energie cinetique de la bille est absorbé lors du rebond, avec la bille qui s'arrette donc relativement au plateau.

Ensuite $\mu \leq 1$ par conservation de l'energie, pour avoir dans le referenciel du plateau (donc du rebond) $E_c^f \leq E_c^i$ qui evite un gain d'energie de la bille lors du rebond. Le cas $\mu = 1$ correspond donc a des collisions elastique avec conservation de l'energie cinetique.

1. Les paramètres de contrôle sont l'amplitude A et la pulsation Ω du vibreur. Proposer un paramètre sans dimension. Par la suite, Ω est fixé et seul A peut varier. Par la suite on prendra $\Omega/2\pi$ = 30 Hz.

Reponse: Ce parametre adimensioné f doit comparer les deux force en jeu: la pesanteur et la force appliqué par le plateau. On fixe donc $f = \frac{A\Omega^2}{g}$.

1. Décrire l'espace des phases.

Reponse: L'espace des phases est à 2 dimensions, que l'on peut representer par $\left(v_n^i, z_n\right)$ ou $\left(v_n^i, t_n\right)$.

1. On suppose le vibreur immobile. La bille est lancée de l'altitude du plateau avec une vitesse v^i. Donner le temps de montée, le temps de vol jusqu'au premier rebond et l'altitude

Reponse: On a pour une chute libre $v(t) = v^i - gt$ et $z(t) = v^i t - \frac{g}{2}t^2$. On trouve donc le temps de monté t_m pour $v(t_m) = 0$ qui nous donne $t_m = \frac{v^i}{g}$. On a alors l'altitude $h = z(t_m)\left(v^i\right)^2 \frac{1}{2g}$. On peut finalement trouver le temps de vol t_v pour $z(t_v) = 0$ qui nous donne $t_v = 2t_m = \frac{2v^i}{g}$.

1. Le vibreur est toujours immobile. Existe-t-il un attracteur ? Si oui, quel est son bassin d'attraction ?

Reponse: Pour $\mu = 1$, on a conservation de la vitesse $v^i = v^f$ pour toute vitesse. Pour $0 \leq \mu < 1$ le seul point fixe sera $v^i = v^f = 0$, et son bassin d'attraction sera l'ensemble de l'espace de phases.

2 - Régimes périodiques

2.1 - Approche analytique

1. On suppose que le mouvement de la bille est de même période $T = 2\pi/\Omega$ que celui du vibreur. Montrer sans calcul que $v_n^i = -v_n^f$, exprimer v^i en fonction de la période T et en déduire la vitesse du vibreur au moment du choc.

Reponse: Le plateau est donc à la meme position à chaque choc, et la bille a donc la meme altitude apres le n^{iem} choc et avant le $(n+1)^{iem}$ choc. L'energie potentiel de pesenteur est donc la meme, et par conservation de l'energie cinetique on a donc $v_n^i = -v_n^f$.

1. Les chocs ont lieu à des instants notés t_n. La variable pertinente est le déphasage φ_n entre l'instant T_{n-1} où le plateau passe à son point haut après le choc $n - l$. φ_n peut donc être supérieur à 2π : $\varphi_n = \Omega(t_n - T_{n-1})$. Expliquer succintement ce qu'elle représente, sachant que ce choix de variable est lié à l'expérience qui utilise la fonction de synchronisation de l'oscilloscope afin de visualiser temporellement les chocs.

Reponse: φ_n represente le dephasage entre le mouvement du plateau et les chocs. Cette grandeur est très visuel si on observe les rebonds avec un osciloscope synchroniser sur les pics de la position du plateau. φ_n apparait alors simplement comme le temps (plus precisement phase) de decalage entre "la gauche de l'ecran" (le dernier signal de synchronisation) et le signal du rebond.

1. Montrer que pour obtenir un mouvement de la bille à la même pulsation que le forçage, l'amplitude du vibreur doit vérifier

$$\mathrm{A} = -\frac{\mathrm{A}_c}{\sin(\varphi_n)} \tag{1}$$

et donner l'expression de A_c et sa valeur numérique.

Reponse: Pour obtenir ce mouvement de la bille, on doit avoir $v_n^f = v_{n+1}^i$. Or $\mathcal{A}(t_n) = \Omega \mathrm{A} \sin(\Omega t_n) = \Omega \mathrm{A} sin(\varphi_n)$. On a donc:

$$v_n^f + \Omega \mathrm{A} \sin(\varphi_n) = -\mu \left[-v_n^f + \Omega \mathrm{A} \sin(\varphi_n) \right]$$

$$v_n^f (1-\mu) = -(1+\mu) \Omega \mathrm{A} \sin(\varphi_n)$$

$$A = -\frac{v^i}{\Omega} \frac{1-\mu}{1+\mu} \frac{1}{\sin(\varphi_n)} = -\frac{\mathrm{A}_c}{\sin(\varphi_n)}$$

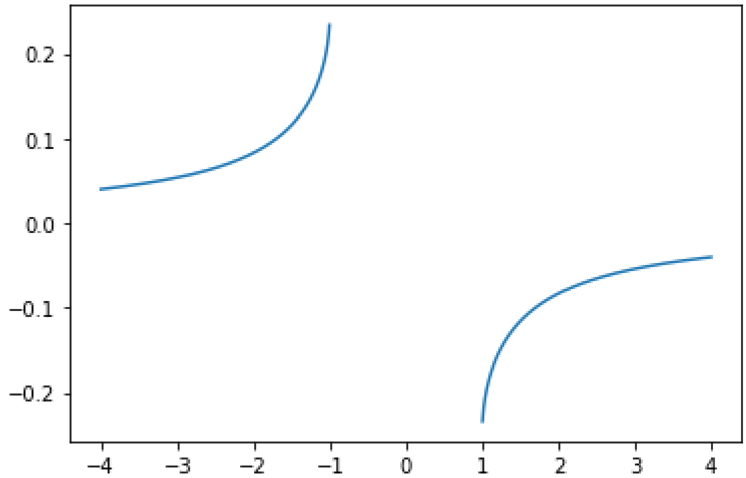
Avec $\mathrm{A}_c = \frac{v^i}{\Omega} \frac{1-\mu}{1+\mu}$.

1. Tracer les solutions de l'expression ci-dessus, c'est à dire $\varphi_n / 2\pi$ en fonction de A. Commenter. Donner la position du point d'amplitude minimale et sa valeur numérique.

Reponse: On fixe $\mathrm{A}_c = 1$ pour simplifier notre graph, on a alors $\varphi_n = \arcsin(-\frac{1}{\mathrm{A}})$ pour A prenant n'importe quel valeur réel:

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt

#%matplotlib inline
#%matplotlib qt

A = np.linspace(-4, 4, 1000)
phi = np.arcsin(-1/A) / (2*np.pi)

plt.plot(A, phi)
plt.show()
```

```
/tmp/ipykernel_16766/4150059644.py:8: RuntimeWarning: invalid value encountered in ar
csin
  phi = np.arcsin(-1/A) / (2*np.pi)
```



On a une amplitude minimal $\mathrm{A} = \mathrm{A}_c$ pour $\varphi_n = \frac{\pi}{2}$, qui correspond au cas ou la bille tape le plateau lorsqu'il a la plus grande vitesse.

1. Reprendre brièvement cette analyse pour le cas où la période du mouvement de la bille est un multiple de T.

Reponse: Si la periode du mouvement de la bille est un multiple de T on a toujours $v_n^i = -v_n^f$ car le plateau est à la meme position à chaque rebond.

Le reste de l'analyse reste inchangée.

1. De manière plus générale - c'est à dire même pour des régimes non periodiques - on propose la section de Poincaré de type stromboscopique qui consiste à échantillonner le mouvement à chaque choc. L'application de Poincaré notée F permet donc de passer de (z_n, v_n^i) à (z_{n+1}, v_{n+1}^i). Par commodité, on préfère étudier la variable φ_n plutôt que z_n. Justifier que ce choix n'induit pas de perte d'information.

Reponse: On a pas de perte d'information car z_n est determinable à partir de φ_n.

1. Comment se présente une trajectoire périodique dans cette section de Poincaré ?

Reponse: Une trajectoire périodique est un point fixe de l'application F. On peut avoir $F(\varphi, v^i) = (\varphi, v^i)$ si chaque rebond est le meme que le precedent, mais on peut aussi avoir plus generalement une trajectoire periodique sur N rebond si $F^N(\varphi, v^i) = (\varphi, v^i)$

1. Ecrire l'équation que vérifie $t_{n+1} - t_n$, intervalle de temps entre les rebonds n et $n+1$. Cette équation n'est pas soluble analytiquement et il faut avoir recours aux outils numériques.

Reponse: On connais l'altitude et la vitesse à t_n, on peut alors decrire la chute libre suivant t_n:

$$z(t) = \mathrm{A}\cos(\varphi_n) + v_n^f(t - t_n) + \frac{g}{2}(t - t_n)^2$$

On à donc la condition $z(t_{n+1}) = \mathcal{A}(t_{n+1})$ qui s'ecrit:

$$\mathrm{A}\cos(\varphi_n) + v_n^f(t_{n+1} - t_n) + \frac{g}{2}(t_{n+1} - t_n)^2 = \mathrm{A}\cos(\varphi_n + \Omega(t_{n+1} - t_n))$$

2.2 - Approche numérique

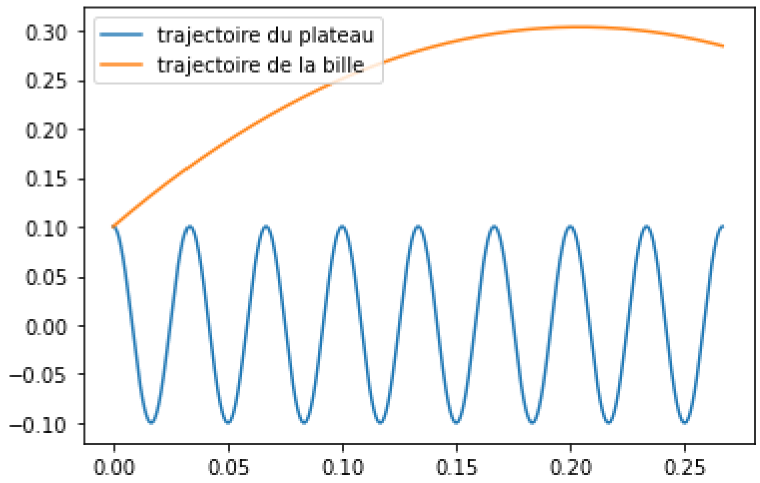
Nous nous intéressons au régime stationnaire de ce système quand il existe. Le mouvement de la bille peut être totalement déterminé en connaissant les instants t_n des rebonds. On conseille de procéder par dichotomie. Décrire brièvement la méthode employée.

1. Ecrire une fonction qui trace et affiche la trajectoire du plateau ainsi que celle d'une bille en chute libre (ignorant le contact avec le plateau pour le moment) entre les instants t_A et t_B en fonction de t_A, t_B, de la vitesse v_A de la bille au temps t_A (en supposant que la bille et le plateau sont à la même altitude au temps t_A), d'un pas de temps dt de discrétisation de ces trajectoires, et de l'amplitude A de l'oscillation du plateau.

In [2]:
```
import numpy as np
from math import pi,cos,sin,acos,atan

#%matplotlib inline
#%matplotlib qt


Freq=30 # [ Hz ]
Omega= 2*pi* Freq
T=1/Freq
mu=0.53
```

In [3]:
```
g=9.81

# question 1
def traj_simple(ta, tb, dt, va, A):
    tmp = np.arange(ta, tb, dt)
    plateau=A*np.cos(Omega*tmp)
```

```
    bille=A*np.cos(Omega*ta) + (tmp - ta)*(va - g/2*(tmp - ta))

    plt.plot(tmp, plateau, label="trajectoire du plateau")
    plt.plot(tmp, bille, label="trajectoire de la bille")
    plt.legend()

traj_simple(0, 8*T, 1e-4, 2, 0.1)
plt.show()
```



1. Ecrire une fonction qui retourne, en fonction du temps de décollage t_n de la bille et de sa vitesse initiale v_n^i et de l'amplitude A de la vibration, le temps du prochain rebonds t_{n+1} et la vitesse de chûte v_{n+1}^f. La fonction utilisera, comme variable ou comme constante globale, une tolérance acceptable dt_d qui fixera l'arrêt de la procédure de dichotomie.

Reponse: On ne peut pas simplement faire une recherche par dichotomie car il peut y avoir plusieur racine très serré, et l'on ne veut trouver que la premiere.

On va plutot utiliser l'algorithme suivant:

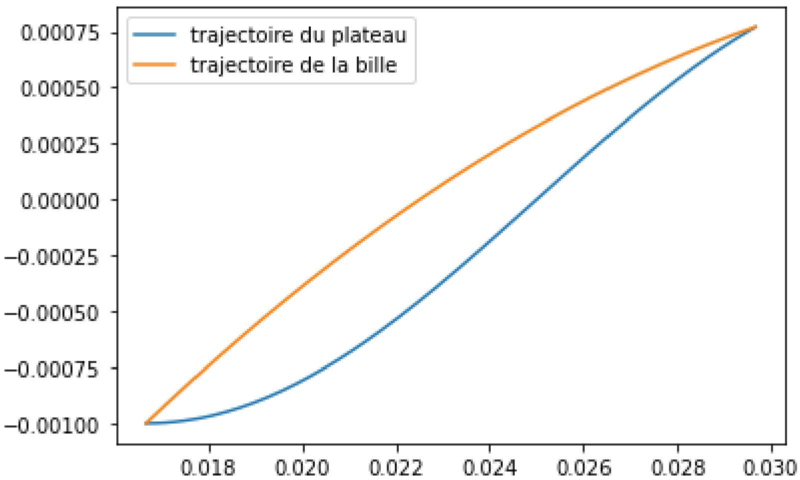
- On fait d'abord un recherche linéaire avec un parametre relativement large dt_large, mais suffisement petit devant la periode pour assurer que plusieur racine ne se trouve pas dans un interval dt_large
- Si l'altitude de la bille depasse l'altitude maximum du plateau, on avance à la fin de son temps de vol que l'on peut facilement calculer.
- Quand on trouve un interval dt_large avec un changement de signe, on fait une recherche par dichotomie avec un tolerance dtd

In [4]:
```
# question 2
dtd = 1e-13
dt_large = 1e-4
max_dichot = 20

def step(tn, vin, A):
    hi = A*cos(Omega*tn)

    # equation à resoudre
    altitude_bille = lambda t : hi + (t - tn)*(vin - g/2*(t - tn))
    f = lambda t : altitude_bille(t) - A*cos(Omega*t)


    # calcul de tn+1=tf
    tf = tn + dt_large
    vol_passe = False
    while f(tf) > 0:
        # passer le vol au dessus du plateau
        if altitude_bille(tf) > A and not vol_passe:
```

```python
            tf += 2*(vin - g*(tf - tn))/g
            vol_passe = True
            continue

        tf += dt_large

    # retour en arriere si besoin
    while f(tf - dt_large) < 0 and tf - dt_large > tn:
        tf -= dt_large

    # recherche par dichotomie plus precise
    if f(tf) < 0:
        a, b = tf - dt_large, tf
        for _ in range(max_dichot):
            # millieu
            millieu = (a + b)/2

            if f(a)*f(tf) < 0:
                a = millieu
            else:
                b = millieu

            if b - a < dtd:
                break
        tf = (a + b)/2

    #vitesse final
    vf = vin - (tf - tn)*g

    return tf, vf
```

1. Vérifier graphiquement que la fonction fonctionne correctement notamment pour les paramètres suvants : $A = 1$ mm, $t_0 = T/2$, $v_0^i = 0.2$ m/s

In [5]:
```python
A = 1e-3
t0 = T/2
vi0 = 0.2
t1, _ = step(t0, vi0, A)
traj_simple(t0, t1, 1e-4, vi0, A)
plt.show()
```



1. Il arrive que la bille progresse par rebonds de très faibles amplitudes, c'est à dire que la quantité $\frac{v_n^f - \dot{\mathcal{A}}(t_n)}{\dot{\mathcal{A}}(t_n)}$ soit petite. Dans ce cas, il est conseillé de considérer que la bille reste collée au vibreur pendant un certain temps. Ecrire la condition qui fixe le temps de redécollage de la bille.

In [6]:
```python
lim_accrochage=0.04
def accrochage(uf, tn):
```

```python
    v_plateau = -Omega*A*sin(Omega*tn)
    return abs((uf - v_plateau)/v_plateau) < lim_accrochage


def t_decollage(t, A):
    # on cherche la condition :
    # a_plateau = -Omega*Omega*A*cos(Omega*t) = -g

    t_max = np.ceil(t/T)*T

    if Omega*Omega*A < g:
        return t_max

    # solution de l'equation
    t0 = acos(g/(Omega*Omega*A))/Omega

    t_min = np.floor(t/T)*T + t0



    while t_min < t:
        t_min += T
    return t_min
```

1. Ecrire une fonction qui donne les instants $\{t_n\}$ et les vitesses $\{v_n^i\}$ et $\{v_n^f\}$ des N premiers rebonds en fonction de N, A,t_0 et de la vitesse initiale de la bille v_0^i. On supposera qu'à l'instant t_0 la bille a la même altitude que le plateau.

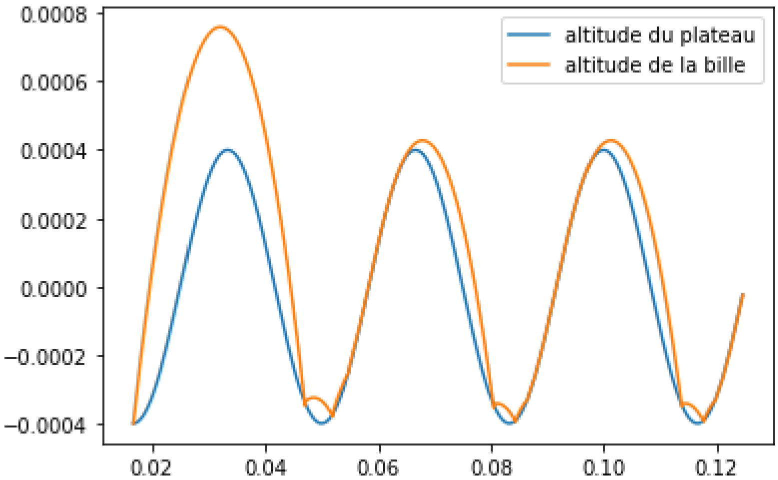
In [7]:
```python
def steps(N, A, t0, vi):
    tn = [t0]
    tdn = [t0] # temps decollage
    vin = [vi]
    vfn = []
    for i in range(N):
        t, vf = step(tdn[i], vin[i], A)

        vfn.append(vf)
        tn.append(t)

        if accrochage(vf, t):
            td = t_decollage(t, A)
            vi = -A*Omega*sin(Omega*td) + A/dt_large

            vin.append(vi)
            tdn.append(td)
        else:
            v_plateau = -Omega*A*sin(Omega*t)
            vi = v_plateau - mu*(vf - v_plateau)

            vin.append(vi)
            tdn.append(t)

    return vin, vfn, tn, tdn
```

Trajectoires

1. Ecrire une fonction qui trace les mouvements de la bille et du plateau pour les N premiers rebonds. Pour quelques valeurs de A, tracer l'altitude de la bille et du vibreur en fonction du temps sur suffisamment de périodes pour voir le régime stationnaire. On propose comme conditions initiales : la bille est posée sur le vibreur avec pour le moment une vitesse égale à deux fois la vitesse maximale du vibreur ($v_0^i = 2\mathrm{A}\Omega$). Pour A=0.4 mm, vérifiez graphiquement l'équation $\mathrm{A} = -\mathrm{A}_c/\sin(\varphi_n)$.

In [8]:
```
def obtenir_trajectoire(N, A, t0, vi, dt):
    vin, vfn, tn, tdn = steps(N, A, t0, vi)

    t = np.arange(t0, tdn[-1], dt)
    h_plateau = A*np.cos(Omega*t)
    h_bille = np.array([h_plateau[0]])

    def find_idx_t(T):
        i_min, i_max = 0, len(t) - 1
        while i_max - i_min > 1:
            i_middle = int((i_min + i_max)/2)
            if t[i_middle] <= T:
                i_min = i_middle
            else:
                i_max = i_middle
        return i_max

    i_ti = 0
    for i in range(N):
        i_tf = find_idx_t(tn[i + 1])
        i_td = find_idx_t(tdn[i + 1])

        h0 = A*np.cos(Omega*tdn[i])
        sub_t = t[i_ti + 1:i_tf + 1] - tdn[i]
        sub_h = h0 + sub_t*(vin[i] - sub_t*g/2)

        h_bille = np.append(h_bille, sub_h, 0)

        # durée d'accrochage au plateau
        h_bille = np.append(h_bille, h_plateau[i_tf + 1: i_td + 1], 0)

        i_ti = i_td

    return t, h_plateau, h_bille

def tracer_trajectoire(N, A, t0, vi, dt):
    t, h_plateau, h_bille = obtenir_trajectoire(N, A, t0, vi, dt)

    plt.plot(t, h_plateau, label="altitude du plateau")
    plt.plot(t, h_bille, label="altitude de la bille")
    plt.legend()
```

In [9]:
```
tracer_trajectoire(100, 4e-4, T/2, 2*4e-4*Omega, 1e-4)
plt.show()
```



In [10]:
```
tracer_trajectoire(20, 5e-4, T/2, 3*4e-4*Omega, 1e-4)
plt.show()
```

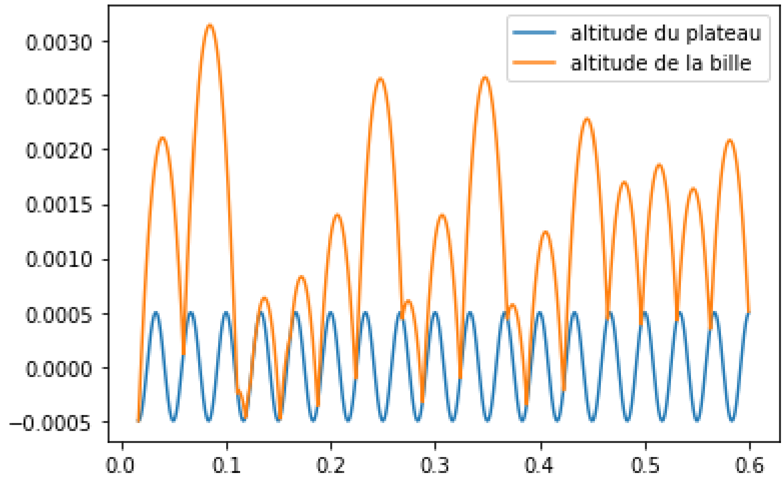


Diagramme de bifurcation

1. Pour résumer les différents comportements observés, tracer le diagramme de bifurcations φ_n en fonction de A. Sur chaque branche, indiquer le rapport T_0/T, où T_0 est la période de la bille et T celle du vibreur. On pourra ne garder que les rebonds pour lesquels $\varphi_n > 0$. Parfois, pour certaines conditions initiales v_0^i certaines solutions stationnaires ne sont pas "trouvées" par la bille. Afin d'obtenir un diagramme de bifurcation un peu complet, il peut être intéressant de lancer plusieurs fois la bille avec des valeurs de v_o^i différentes.

Reponse: On definit d'abord un fonction qui retourne la phase en fonction du temps.

On peut ensuite detecter la periodicité d'une solution, en cherchant avec un tolerance, deux point ayant la meme phase φ_n et la même vitesse v_n^i. On s'assure que ces deux point ne sont pas plus proche que une periode, et pas plus eloigner que un certain nomber de periodes (pour des raisons de stabilité).

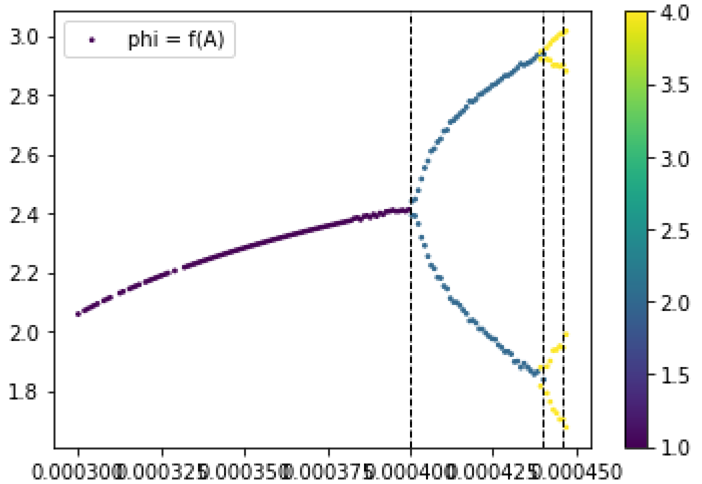
On peut ensuite tracer le diagrame de bifurcation. Notez que la fonction "obtenir_bifurc" print une liste des "dephasage" (periode en nombre de rebond) trouvé, pour pouvoir suivre son avancé.

In [11]:
```
# fonction qui nous donne phin en fonction de
def phi_n(t):
    t_min = np.floor(t/T)*T
    return (t - t_min)*Omega - pi

# fonction qui nous donne le phin de stabilité obtenue par une suitte de tn
phi_tol = pi*0.02 # tolerance entre phi_n et phi_{n + 1} pour considérer que l'on a s
min_t_var = T*0.9
max_t_var = T*32.1
vi_relative_tol = 0.02
max_dephasage = 32
max_dephasage_par_periode = 3
def phi_n_stabilite(vin, tn):
    v_target, phi_target = vin[-1], phi_n(tn[-1])
    for dephasage in range(1, min(len(vin) - 1, max_dephasage + 1)):
        if tn[-1] - tn[-1 - dephasage] < min_t_var:
            continue
        elif dephasage*T/(tn[-1] - tn[-1 - dephasage]) > max_dephasage_par_periode:
            return False, 0

        if tn[-1] - tn[-1 - dephasage] > max_t_var:
            return False, 0

        v, phi = vin[-1 - dephasage], phi_n(tn[-1 - dephasage])
        if abs(v - v_target) < vi_relative_tol*abs(v + v_target)/2 and abs(phi - phi_
            return True, dephasage

    return False, 0
```

In [12]:
```
def obtenir_bifurc(Amin, Amax, NA, vimin, vimax, Nv, N):
    print("dephasages trouvés: ", end="")
    A_stable, phin, T_T0 = [], [], []
    for A in np.linspace(Amin, Amax, NA):
        for v in np.logspace(np.log10(vimin), np.log10(vimax), num=Nv):
            vin, vfn, tn, tdn = steps(N, A, T/2, v)

            stable, dephasage = phi_n_stabilite(vin, tn)
            if stable:
                print(f"{dephasage}, ", end="")

                A_stable += [A] * dephasage
                T_T0 += [(tn[-1] - tn[-dephasage - 1])/T] * dephasage
                phin += [phi_n(t) for t in tn[-dephasage:]]
                break

    return A_stable, phin, T_T0

A = 3e-4
A_max=4.4705e-4
vi = 2*A*Omega
A_stable, phin, T_T0 = obtenir_bifurc(A, A_max, 153, vi*1e-1, vi*1e4, 6, 600)
```

```
dephasages trouvés: 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 4, 2, 4, 4, 4, 4, 4, 4, 4,
```

In [13]:
```
# tracer bifurcation
plt.scatter(A_stable, phin, marker=".", s=12, c=T_T0, label=f"phi = f(A)")
plt.colorbar()

# tracer les A_i
A_i = [4e-4, 4.4e-4, 4.46e-4]
y_limits = plt.gca().get_ylim()
for a in A_i:
    plt.plot([a, a], y_limits, "k--", linewidth=1)
plt.gca().set_ylim([max(1.4, y_limits[0]), y_limits[1]])

plt.legend()
plt.show()
```



1. On nomme A_q l'amplitude à l'apparition de la bifurcation numéro q (La valeur de A_1 sera calculée explicitement dans la partie 2.3).

Calculer le rapport $\delta = \frac{A_{q+1} - A_q}{A_q - A_{q-1}}$ et le commenter.

Reponse: On a marqué les bifurcation facilement visible par des trait verticaux. On obtient les valeurs $A_1 = 0.4mm$, $A_2 = 0.44mm$ et $A_3 = 0.446mm$.

On note que la n^{iem} bifurcation correspond à 2^n branche avec une periode $T_0/T = 2^n$ (representé sur le diagram par la couleur des points).

En posant $\delta_q = \frac{\mathrm{A}_{q+1}-\mathrm{A}_q}{\mathrm{A}_q-\mathrm{A}_{q-1}}$ on obtient la valeurs $\delta_2 = 0.15 = \frac{3}{20}$.

En supposant $\delta_q = \delta = 0.15$ constant. On a alors $\mathrm{A}_{q+1} - \mathrm{A}_q = \delta^{q-1}\left(\mathrm{A}_2 - \mathrm{A}_1\right)$ Et donc:

$$\mathrm{A}_q = A_1 + \left(\mathrm{A}_2 - \mathrm{A}_1\right)\sum_{i=0}^{q}\delta^i = A_1 + \left(\mathrm{A}_2 - \mathrm{A}_1\right)\frac{1-\delta^q}{1-\delta}$$

Et on a donc:

$$\lim_{q \to +\inf} \mathrm{A}_q = A_1 + \frac{\mathrm{A}_2 - \mathrm{A}_1}{1-\delta}$$

Avec les valeur experimentales trouvé on obtient $\lim_{q\to+\inf} \mathrm{A}_q = 4.4705.10^{-4}$, que l'on a utilisé comme limite du diagram de bifurcation.

2.3 Etude de la stabilité pour un mouvement de période T

L'étude conduite dans la sous-partie 2.1 prévoit des branches qui n'apparaissent pas lors des simulations numériques. On peut donc légitimement soupçonner qu'elles sont instables. Pour déterminer la stabilité d'une trajectoire périodique, on utilise sa matrice de monodromie définie par

$$\begin{pmatrix} \delta\varphi_{n+1} \\ \delta v^i_{n+1} \end{pmatrix} = M \begin{pmatrix} \delta\varphi_n \\ \delta v^i_n \end{pmatrix} \tag{2}$$

où $\delta\varphi_n = \varphi_n - \varphi_p$ et $\delta v^i_n = v^i_n - v^i_p$ (on a noté φ_p et v^i_p les valeurs de φ et v^i pour le mouvement périodique) sont des petits écarts à la trajectoire périodique. Pour la trajectoire de période T, la matrice de monodromie s'écrit

$$M = \begin{pmatrix} 1 & \frac{\Omega}{g}(1+\mu) \\ \frac{g}{\Omega}\pi(1-\mu)\tan^{-1}(\varphi_T) & \mu^2 + \pi(1-\mu^2)\tan^{-1}(\varphi_T) \end{pmatrix} \tag{3}$$

1. *Bonus* Expliquer comment on obtient cette matrice.

Reponse: Soit $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ on cherche les valeur de a, b, c et d au premier ordre en $\delta\varphi_n$ et δv^i_n autour d'une trajectoire periodique. On les obtient en etudiant les variation sur un rebond:

- a vaut 1 car $\delta\varphi_n$ decale φ_n et donc directement φ_{n+1}.
- c On a deux termes: μ^2 viens simplement du gain de vitesse après deux rebond. $\pi(1-\mu^2)\tan^{-1}(\varphi_T)$ correspond à l'impact sur le rebond de la variation de vitesse du plateau causé par la variation de temps de vol causé par δv^i_n.
- b est la variation de phase causé par la variation de vitesse apres rebond qui engendre une difference de temps de vol.
- d est la variation de vitesse causé par la difference de vitesse du plateau causé par le decalage de la phase.

1. Que vaut $\det(M)$? Que représente-t-il ?

Reponse: On trouve facilement $det(M) = \mu^2 \leq 1$.

Or $det(M)$ etant le produit des valeurs propre il represente la dilatation/contraction du flow. On a donc ici $0 \leq det(M) \leq 1$, donc dans la limite $\mu = det(M) = 1$ on a un flow conservatif, et sinon on a contraction du flow.

1. Montrer qu'une des valeurs propres de M est toujours supérieure à 1 si $\tan^{-1}(\varphi_T) > 0$. En déduire qu'une des branches est instable.

Reponse: Soit λ_1, λ_2 les deux valeur propres de M, et λ'_1, λ'_2 les valeur propres de $M - I$ on a $\lambda_1 = \lambda'_1 + 1$ $det(M-I) = \lambda'_1 * \lambda'_2$. Or on trouve facilement que pour $\tan^{-1}(\varphi_T) > 0$:

$$det(M-I) = -\pi\left(1-\mu^2\right)\tan^{-1}(\varphi_T) < 0$$

On a donc $\lambda'_1 * \lambda'_2 < 0$ et on a donc une valeur propre de $M - I$ negative et une positive. On a alors la valeur propre positive de $M - I$ qui correspond à une valeur propre supérieure à 1 de M.

On a donc toujours une valeur propre de M supérieure à 1 si $\tan^{-1}(\varphi_T) > 0$. Cette branche est donc instable, car on a croissance (et divergance) de la perturbation autour de la trajectoire periodique.

1. Montrer que la valeur propre -1 est obtenue pour $\tan(\varphi_T) = -\frac{\pi}{2}\frac{1-\mu^2}{1+\mu^2}$. Calculer la valeur de A correspondante. En déduire les zones de stabilité et d'instabilité pour l'autre branche et déterminer le seuil d'apparition du doublement de période.

Reponse: Pour que la valeur propre -1 soit obtenue, il faut que $det(M+I) = 0$. Par la meme formule que precedement on trouve facilement que:

$$0 = det(M+I) = 2 + 2\mu^2 + \pi\left(1-\mu^2\right)\tan^{-1}(\varphi_T)$$

Ce qui nous donne trivialement $\tan(\varphi_T) = -\frac{\pi}{2}\frac{1-\mu^2}{1+\mu^2}$.

On a alors doublement de periode pour la valeur propre $\lambda = -1$ obtenue pour $\varphi_T = \arctan\left(-\frac{\pi}{2}\frac{1-\mu^2}{1+\mu^2}\right)$.

Or on a stabilité de la branche pour une valeur propre $\lambda \in \left]-1, 1\right[$ qui correspond à $\varphi_T \in \left]\arctan\left(-\frac{\pi}{2}\frac{1-\mu^2}{1+\mu^2}\right), 0\right[$

1. Comparer avec les simulations numériques et commenter.

Reponse: On a donc avec $\mu = 0.53$ $\varphi_T = \arctan\left(-\frac{\pi}{2}\frac{1-\mu^2}{1+\mu^2}\right) = -0.72 rad$.

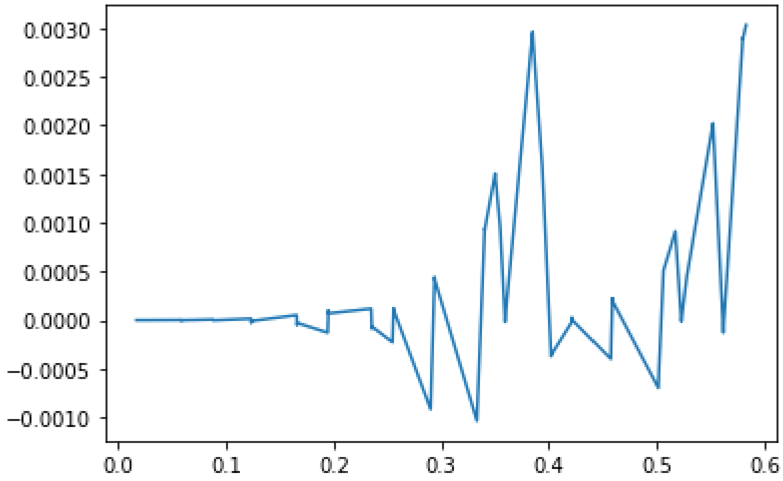
3 - Chaos et attracteur étrange

1. Pour A=0.5 mm, par exemple, montrer la sensibilité aux conditions initiales en traçant l'écart entre les z_n en fonction du temps de deux trajectoires initialement très proches.

Reponse: On peut observer sur le plot que la distance entre les deux trajectoire grandit, prouvant l'instabilité de ce probleme.

In [14]:
```
times, h_plateau, traj1 = obtenir_trajectoire(90, 5e-4, T/2, 1.5e-2, 1e-4)
_, _, traj2 = obtenir_trajectoire(90, 5e-4, T/2, 1.4e-2, 1e-4)

if len(traj1) < len(traj2):
    traj2 = traj2[:len(traj1)]
```

```
if len(traj1) > len(traj2):
    traj1 = traj1[:len(traj2)]
    times = times[:len(traj2)]

delta_traj = np.array(traj1) - np.array(traj2)

plt.plot(times, delta_traj)
plt.show()
```



1. Dans la zone chaotique, comment se présente une trajectoire typique (A fixée et suffisamment grande) sur le graphe φ_n en fonction de A ?

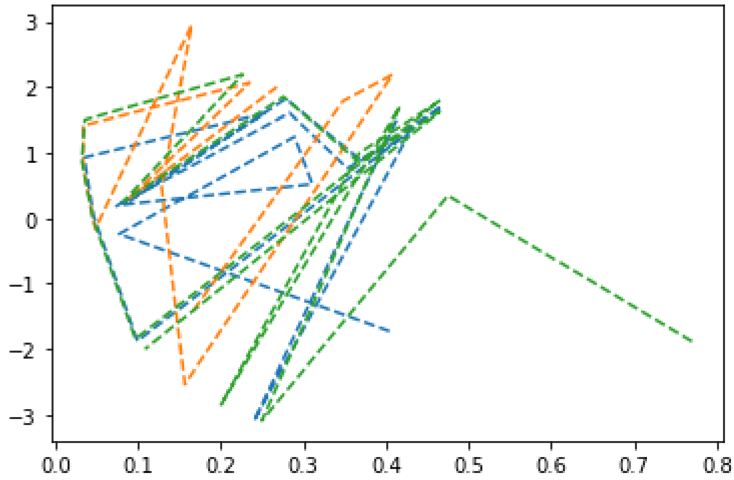
Reponse: Ayant A fixé, on aura un ensemble de point sur la meme ligne verticale correspondant à A. Ce nuage de point sera "dense", mais poura eviter des regions de l'interval possible de φ_n (car dans la zone chaotique la trajectoire effectue des saut de φ_n).

1. Pour obtenir plus d'informations sur les trajectoires, on propose d'utiliser la représentation φ_n en fonction de $v_n^i - v_n^f$. La quantité $v_n^i - v_n^f$ est liée à l'intensité du choc qui peut être mesurée directement par l'amplitude de l'impulsion électrique fournie par la céramique piezo. Quel est l'aspect des trajectoires périodiques dans cette représentation ?

Reponse: Une trajectoire periodique dans cette representation sera simplement un tracé fermé (qui revient sur lui même), bien que la nature discrete de la trajectoire peut engendrer des "saut" (formant une trajectoire qui a des "trous").

1. Tracer une trajectoire typique pour A=0.5 mm, par exemple.

In [32]:
```
def tracer_phase(A, t0, vi, N):
    vin, vfn, tn, tdn = steps(N, A, t0, vi)
    vfn = np.array(vfn)
    vin = np.array(vin[:-1])

    deltaV = vin - vfn
    phase = [phi_n(t) for t in tdn[:-1]]

    plt.plot(deltaV, phase, "--")

A = 5e-4
vi = 2*A*Omega
tracer_phase(A, T/4*0.9, vi, 18)
tracer_phase(A, T/4*1.1, vi/2, 12)
tracer_phase(A, T/4*0.8, vi*2, 17)
plt.show()
```



5- Pourquoi cette structure porte-t-elle le nom d'attracteur ?

Reponse: On peut voir que des trajectoire completement distinctes se rejoigne en tendant vers cette "attracteur" (d'où ce nom).

6- Monter numériquement que cette structure présente un aspect fractal. Cet aspect lui vaut le nom d'attracteur étrange.

Reponse:

In [ ]:

7- Proposer un moyen de calculer numériquement la dimension fractale de cet attracteur étrange.

Reponse: On peut chercher la dimension de boite de cette structure.

Pour cela on pourrait d'abord ecrire une fonction qui detecte si une trajectoire tend vers l'attracteur, en regardant si les point au bout de la trajectoire se rapporche d'une section d'une trajectoire que l'on a seletionner car elle appartient à l'attracteur.

On peut ensuite quadriller l'espace des phase avec des point de depart ecarté d'un parametre ϵ, et compter le nombre de point $n(\epsilon)$ qui tendent vers l'attracteur.

On peut ensuite iterativement raffiner le quadrillage, et meusurer la valeur limite du ratio $\frac{\ln(n(\epsilon))}{-\ln(\epsilon)}$ qui sera la dimension de ce fractale.

8- *Bonus* Donner sa valeur.

9- Commenter le rôle de μ.

Reponse: μ est un parametre de controle continue, qui est un parametre d'atenuation temporelle de $v_n^i - v_n^f$.

On pourait s'attendre à ce que le "rayon" de l'atracteur dans la dimension $v_n^i - v_n^f$ reduise avec μ.

In [ ]: