Sam Kendig
6.946j/8.351j/12.620j - Structure and Interpretation of Classical Mechanics
Problem Set 6

- 3.8: Sleeping top

  $U_p(\theta) = \frac{p^2}{2C} - \frac{p^2}{2A}\tan^2\frac{\theta}{2} + gMR\cos\theta$

  $DU_p(\theta) = \frac{p^2}{2A}\frac{\sin\frac{\theta}{2}}{\cos^3\frac{\theta}{2}} - gMR\sin\theta$

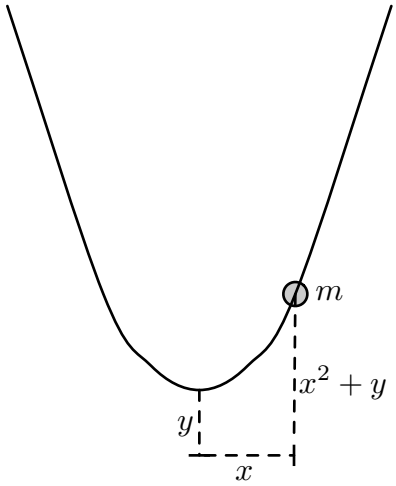
  $D^2U_p(\theta) = \frac{3p^2}{4A}\frac{1}{\cos^4\frac{\theta}{2}} - \frac{p^2}{2A}\frac{1}{\cos^2\frac{\theta}{2}} - gMR\cos\theta$

  $D^2U_P(0) = \frac{p_c^2}{4A} - gMR = 0$

  $\omega_c = \frac{p_c}{C} = \frac{\sqrt{4gmRA}}{C}$

- 3.10: Fun with phase portraits

  Consider a bead of mass m on a parabolic wire being displaced vertically, as shown below:

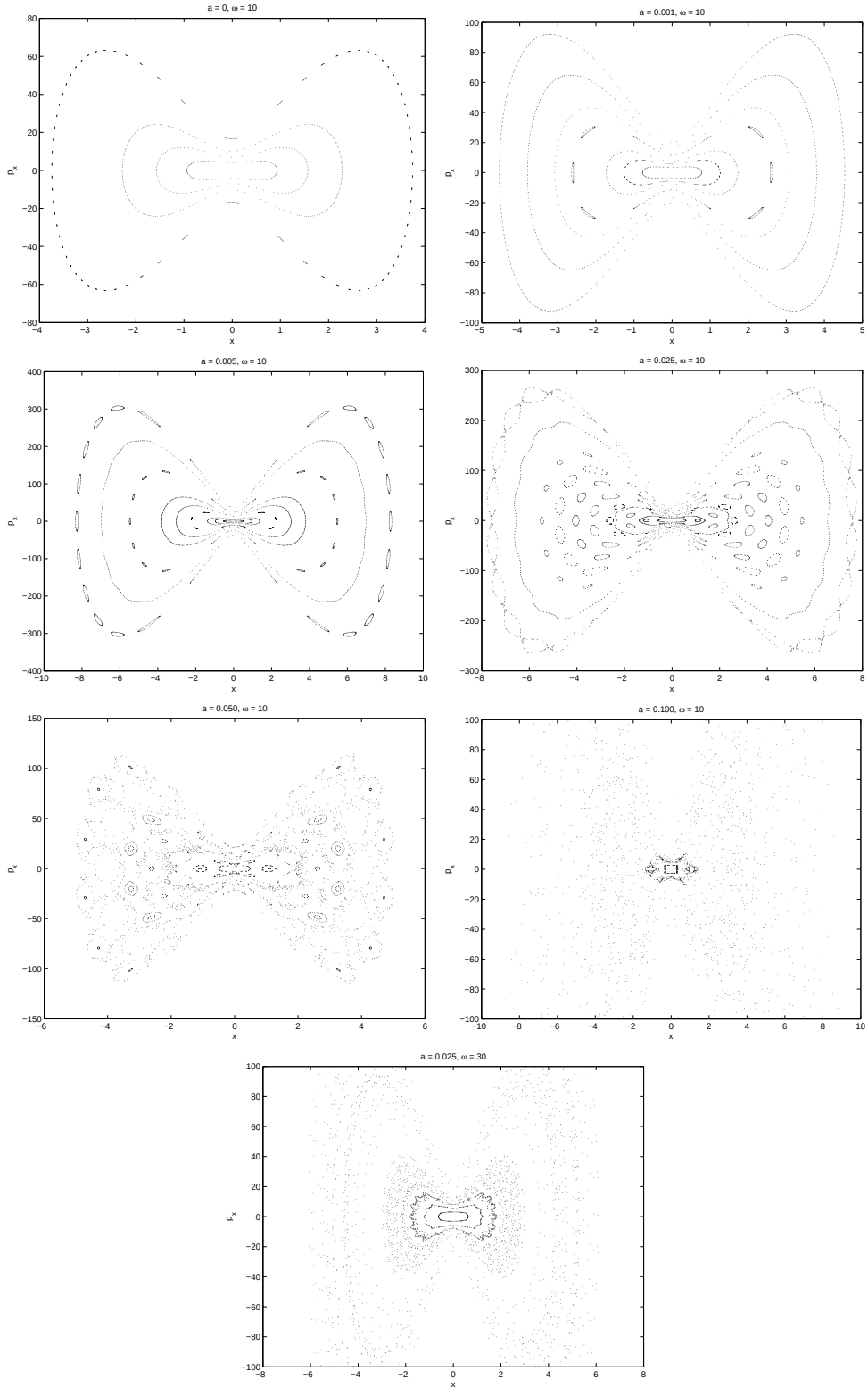


  $T = \frac{1}{2}m(\dot{x}^2 + (2x\dot{x} + \dot{y})^2)$
  $V = mg(x^2 + y)$

  $L(t, x, \dot{x}) = \frac{1}{2}m(\dot{x}^2 + (2x\dot{x} + \dot{y})^2) - mg(x^2 + y)$

```
(define ((L-para m g y) local)
  (let* ((t (time local))
 (x (coordinate local))
 (xdot (velocity local))
```

```
 (vy (D y)))
    (- (* 1/2 m (+ (square xdot)
   (square (+ (* 2 x xdot)
      (vy t)))))
       (* m g (+ (square x) (y t))))))

(define ((periodic-drive a omega phi) t)
  (* a (cos (+ (* omega t) phi))))

(define (L-para-driven m g a omega)
  (L-para m g (periodic-drive a omega 0)))

(define (H-para-sysder m g a omega)
  (Hamiltonian->state-derivative
   (Lagrangian->Hamiltonian
    (L-para-driven m g a omega))))

(define (para-map m g a omega)
  (let ((advance (state-advancer H-para-sysder m g a omega))
(map-period (/ :2pi omega)))
    (lambda (x px return fail)
      (let ((ns (advance
 (up 0 x px)
 map-period)))
(if (> (abs x) 10)
    (fail)
    (return (coordinate ns)
    (velocity ns)))))))

(define win (frame -10 10 -50 50))


(start-gnuplot "para025.dat")

(let ((m 1.0)
      (g 9.8)
      (a 0.025)
      (omega 10))
  (explore-map win
       (para-map m g a omega)
       100))

(stop-gnuplot)
```


a = 0, ω = 10
a = 0.001, ω = 10
a = 0.005, ω = 10
a = 0.025, ω = 10
a = 0.050, ω = 10
a = 0.100, ω = 10
a = 0.025, ω = 30
x
p_x

- 3.13:

  $x' = x\cos\alpha - (y - x^2)\sin\alpha$
  $y' = x\sin\alpha - (y - x^2)\cos\alpha$

  a. Show that the map preserves area:

  $$D\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{bmatrix} \cos\alpha + 2x\sin\alpha & -\sin\alpha \\ \sin\alpha - 2x\cos\alpha & \cos\alpha \end{bmatrix}$$

  $$Jac\begin{pmatrix} x' \\ y' \end{pmatrix} = \cos^2\alpha + 2x\sin\alpha\cos\alpha + \sin^2\alpha - 2x\cos\alpha\sin\alpha = 1$$

  b.

```
(define ((test-map alpha) x y return failure)
  (let ((xp (- (* x (cos alpha))
       (* (- y (square x)) (sin alpha))))
(yp (+ (* x (sin alpha))
       (* (- y (square x)) (cos alpha)))))
    (if (or (> (abs x) 1) (> (abs y) 1))
(failure)
(return xp yp))))

(define window (frame -1 1 -1 1))


(start-gnuplot "quad1.32.dat")
(explore-map window (test-map 1.32) 2000)
(stop-gnuplot)
```

c.