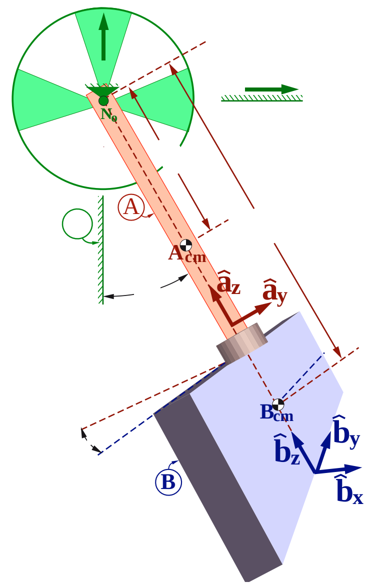
Motivating example (MIPSI): Babyboot



Model

Shown right is a model of a swinging babyboot (uniform plate B) attached by a shoelace (thin uniform rigid rod A) to a fixed rigid support N. Rod A is attached to N by a revolute joint at point N_o of N. B is attached to A with a 2^{nd} revolute joint at point B_o so B can rotate freely about A's axis. Note: The revolute joints' axes are ***perpendicular***, not parallel.

- **Bodies**: The rod and plate are ______ (inflexible/undeformable).
- **Connections**: Revolute joints are ______ (massless, frictionless, no slop).
- **Forces**: ______ is uniform and constant. Other contact forces (e.g., ______ and solar/light pressure) and distance forces (e.g., electromagnetic and other gravitational) are negligible.
- **Newtonian reference frame**: Earth.

Identifiers

Right-handed sets of unit vectors $\hat{\mathbf{n}}_x, \hat{\mathbf{n}}_y, \hat{\mathbf{n}}_z$; $\hat{\mathbf{a}}_x, \hat{\mathbf{a}}_y, \hat{\mathbf{a}}_z$; $\hat{\mathbf{b}}_x, \hat{\mathbf{b}}_y, \hat{\mathbf{b}}_z$ are fixed in N, A, B, respectively, with $\hat{\mathbf{n}}_x = \hat{\mathbf{a}}_x$ parallel to the revolute axis joining A to N, $\hat{\mathbf{n}}_z$ vertically-upward, $\hat{\mathbf{a}}_z = \hat{\mathbf{b}}_z$ parallel to the rod's long axis (and the revolute axis joining B to A), and $\hat{\mathbf{b}}_z$ perpendicular to plate B.

Quantity	Symbol	Type	Value
Earth's gravitational constant	g	Constant	9.81 m/s^2
Distance between N_o and A_{cm}	L_A	Constant	7.5 cm
Distance between N_o and B_{cm}			20 cm
Mass of A			0.01 kg
Mass of B			0.1 kg
A's moment of inertia about A_{cm} for $\hat{\mathbf{a}}_x$	I^{A}	Constant	0.05 kg∗cm^2
B's moment of inertia about B_{cm} for $\hat{\mathbf{b}}_x$	$\mathrm{I}^{\mathrm{B}}_x$	Constant	2.5 kg∗cm^2
B's moment of inertia about B_{cm} for $\hat{\mathbf{b}}_y$		Constant	0.5 kg∗cm^2
B's moment of inertia about B_{cm} for $\hat{\mathbf{b}}_z$		Constant	2.0 kg∗cm^2
Angle from $\hat{\mathbf{n}}_z$ to $\hat{\mathbf{a}}_z$ with $^+\hat{\mathbf{n}}_x$ sense			Varies
Angle from $\hat{\mathbf{a}}_y$ to $\hat{\mathbf{b}}_y$ with $^+\hat{\mathbf{a}}_z$ sense			Varies
Time	t	Independent variable	Varies

Note: Instructor worksheet at www.MotionGenesis.com ⇒ Textbooks ⇒ Resources completes the blanks above.

Physics

Physics from www.MotionGenesis.com ⇒ Get Started ⇒ Chaotic Pendulum (Babyboot).

The ODEs (ordinary differential equations) governing the motion of this mechanical system are[4]

$$\ddot{q}_A = \frac{2\,\dot{q}_A\,\dot{q}_B\,\sin(q_B)\,\cos(q_B)\,(\mathrm{I}^{\mathrm{B}}_x - \mathrm{I}^{\mathrm{B}}_y) - (\mathrm{m}^{\mathrm{A}}\,L_A + \mathrm{m}^{\mathrm{B}}\,L_B)\,g\,\sin(q_A)}{\mathrm{I}^{\mathrm{A}} + \mathrm{m}^{\mathrm{A}}\,L_A^2 + \mathrm{m}^{\mathrm{B}}\,L_B^2 + \mathrm{I}^{\mathrm{B}}_x\cos^2(q_B) + \mathrm{I}^{\mathrm{B}}_y\sin^2(q_B)}$$

$$\ddot{q}_B = \frac{-\dot{q}_A^2\,\sin(q_B)\,\cos(q_B)\,(\mathrm{I}^{\mathrm{B}}_x - \mathrm{I}^{\mathrm{B}}_y)}{\mathrm{I}^{\mathrm{B}}_z}$$

[4] Four methods to form equations of motion are: ***Free-body diagrams*** of A and B (which is inefficient as it introduces up to 10 unknown force/torque measures); D'Alembert's method (***MG road-maps*** of Section 23.1) which efficiently forms the two equations shown for $\ddot{q}_A$ and $\ddot{q}_B$ (but requires a clever selection of systems, points, and unit vectors); ***Lagrange's equations*** (an energy-based method that automates D'Alembert's cleverness); ***Kane's equations*** (a modern efficient blend of D'Alembert and Lagrange).

Simplify and solve

Computers has revolutionized the solution of differential equations. There are many numerical algorithms for solving nonlinear, coupled, variable coefficient, ODEs (ordinary differential equations) including Euler's method, predictor-corrector, Runga-Kutta, etc. In addition, there are many programs (MATLAB®, **M**otion**G**enesis, WolframAlpha, etc.) that make it easy to solve ODEs.

Computer (numerical) solution of ODEs with MotionGenesis (with plotting)

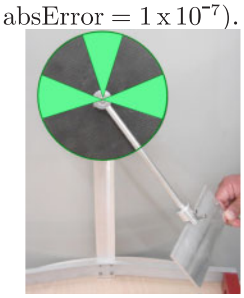
```
Variable  qA'', qB''      % Angles and first/second time-derivatives.
%--------------------------------------------------------------------------
qA'' = 2*( 508.89*sin(qA) - sin(qB)*cos(qB)*qA'*qB' ) / (-21.556 + sin(qB)^2)
qB'' = -sin(qB)*cos(qB)*qA'^2
%--------------------------------------------------------------------------
Input  tFinal = 10 sec, tStep = 0.02 sec, absError = 1.0E-07
Input  qA = 90 deg,  qB = 1.0 deg,  qA' = 0.0 rad/sec,  qB' = 0.0 rad/sec
Output  t sec,  qA degrees,  qB degrees
%--------------------------------------------------------------------------
ODE() solveBabybootODE
Plot solveBabybootODE.1[ 1, 2 ]  % Plot qA (degrees) vs time t (seconds).
Plot solveBabybootODE.1[ 1, 3 ]  % Plot qB (degrees) vs time t (seconds).
Quit
```

Interpret

This simple system has strange non-intuitive motion. For certain initial values of q_A, plate B's motion is well-behaved and "stable" whereas for other initial values of q_A, B's motion is **"chaotic"** – meaning that a small variation in the initial value of q_B or numerical integration inaccuracies lead to dramatically different results (these ODEs test numerical integrators – the plots below required an integrator error of absError = 1 x 10^-7^).

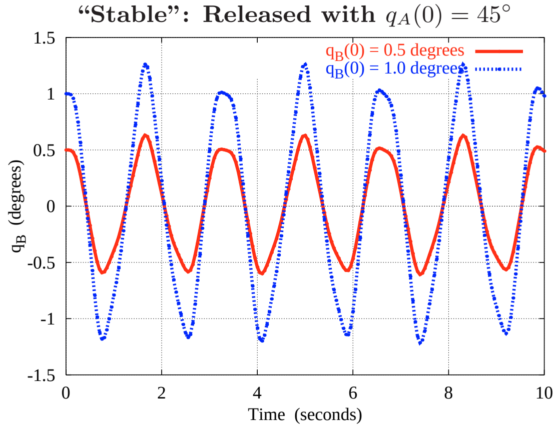
The chart below and figure to the right shows this system's regions of stability (**black**) and instability (***green***).

Initial value of q_A	Stability	
$0° \le q_A(0) \le 71.3°$	**Stable**	**black**
$\mathbf{71.4° \le q_A(0) \le 111.77°}$	***Unstable***	***green***
$111.78° \le q_A(0) \le 159.9°$	**Stable**	**black**
$\mathbf{160.0° \le q_A(0) \le 180.0°}$	***Unstable***	***green***

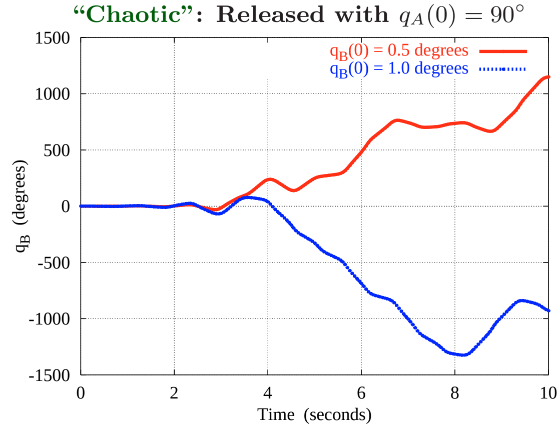


The **"chaotic"** plot below shows q_B is ***very*** sensitive to initial values. A 0.5° change in the initial value $q_B(0)$ results in a $2000^{+°}$ difference in $q_B(t = 10)$!

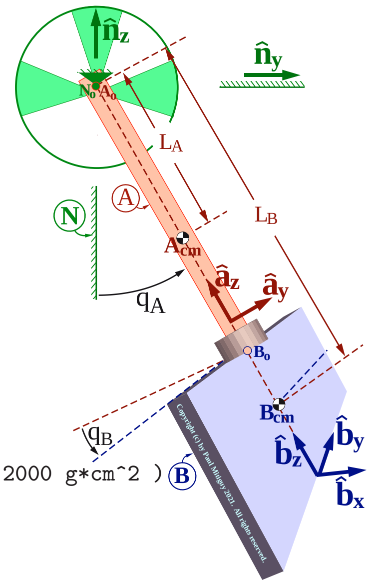
"Stable": Released with $q_A(0) = 45°$

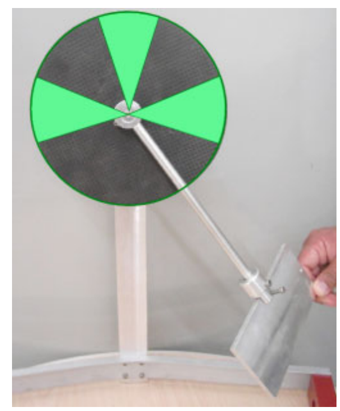


"Chaotic": Released with $q_A(0) = 90°$



More information about this problem is in "Mechanical Demonstration of Mathematical Stability and Instability", *International Journal of Engineering Education (Journal of Mechanical Engineering Education)*, Vol. 2, No. 4, 1974, pp. 45-47, by Thomas Kane. Or visit **www.MotionGenesis.com ⇒ Get Started ⇒ Chaotic Pendulum (Babyboot)**.

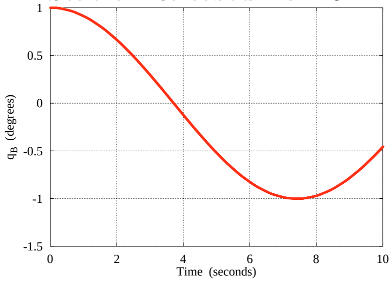




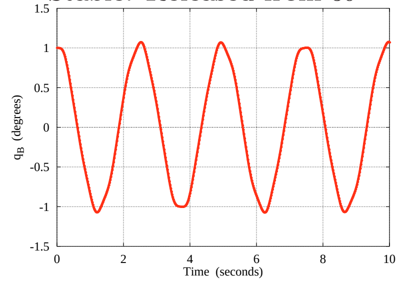
```
% MotionGenesis file:  MGBabybootDynamics.txt
% Problem: Analysis of 3D chaotic double pendulum.
% Copyright (c) 2009 Motion Genesis LLC.  All rights reserved.
%--------------------------------------------------------------------
SetDigits( 5 )               % Number of digits displayed for numbers.
%--------------------------------------------------------------------
NewtonianFrame  N            % Earth.
RigidBody       A            % Upper rod.
RigidBody       B            % Lower plate.
%--------------------------------------------------------------------
Variable   qA''              % Pendulum angle and its time-derivatives.
Variable   qB''              % Plate angle and its time-derivative.
Constant   LA = 7.5 cm       % Distance from pivot to A's mass center.
Constant   LB = 20 cm        % Distance from pivot to B's mass center.
Constant   g = 9.81 m/s^2    % Earth's gravitational acceleration.
A.SetMassInertia( mA =  10 grams,  IAx = 50 g*cm^2,  IAy,  IAz )
B.SetMassInertia( mB = 100 grams,  IBx = 2500 g*cm^2, IBy = 500 g*cm^2, IBz = 2000 g*cm^2 )
%--------------------------------------------------------------------
%   Rotational kinematics.
A.RotateX( N, qA )
B.RotateZ( A, qB )
%--------------------------------------------------------------------
%   Translational kinematics.
Acm.Translate( No, -LA*Az> )
Bcm.Translate( No, -LB*Az> )
%--------------------------------------------------------------------
%   Add relevant contact/distance forces.
System.AddForceGravity( -g*Nz> )
%--------------------------------------------------------------------
%   Equations of motion via free-body-diagrams (MG road-maps).
Dynamics[1] = Dot( Ax>,  System(A,B).GetDynamics(No)   )
Dynamics[2] = Dot( Bz>,            B.GetDynamics(Bcm) )
%--------------------------------------------------------------------
%   Kinetic and potential energy.
KE = System.GetKineticEnergy()
PE = System.GetForceGravityPotentialEnergy( -g*Nz>, No )
MechanicalEnergy = KE + PE
%--------------------------------------------------------------------
%   Optional: Equations of motion with Kane's method.
SetGeneralizedSpeed( qA', qB' )
KaneDynamics = System.GetDynamicsKane()
isSameDynamics = IsSimplifyEqual( Dynamics, KaneDynamics )
%--------------------------------------------------------------------
%   Optional: Equations of motion with Lagranges's method.
SetGeneralizedCoordinates( qA, qB )
LagrangeDynamics = System.GetDynamicsLagrange( SystemPotential = PE )
isSameDynamics := IsSimplifyEqual( Dynamics, LagrangeDynamics )
%--------------------------------------------------------------------
%   Solve dynamics equations for qA'', qB''.
Solve( Dynamics = 0,   qA'',  qB'')
%--------------------------------------------------------------------
%   Integration parameters and initial values.
Input  tFinal = 10 sec,  tStep = 0.02 sec,  absError = 1.0E-07,  relError = 1.0E-07
Input  qA = 90 deg,  qA' = 0.0 rad/sec,  qB = 1.0 deg,  qB' = 0.0 rad/sec
%--------------------------------------------------------------------
%   List output quantities and solve ODEs.
Output  t sec,  qA deg,  qB deg,  MechanicalEnergy Joules
ODE()  MGBabybootDynamics
%--------------------------------------------------------------------
%   Record input together with responses
Save MGBabybootDynamics.html
Quit
```

Investigation of stability: More simulation results

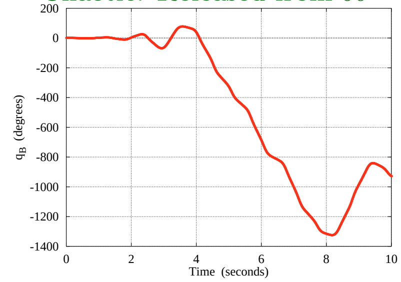
Stable: Released from 5°

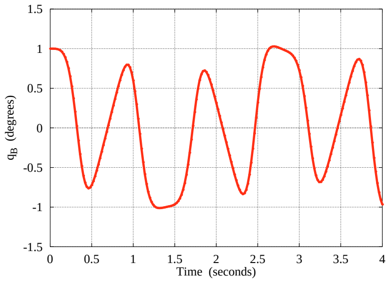


Stable: Released from 30°

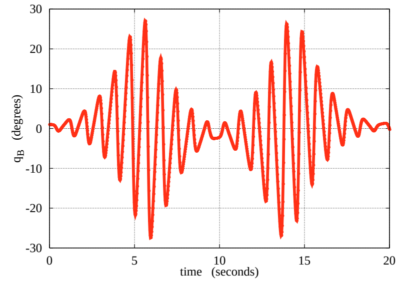


Chaotic: Released from 90°

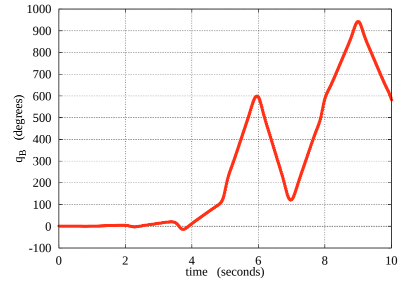




Stable: Released from 135°



Beat: Released from 158°



Chaotic: Released from 177°