Bouncing Ball System
AMATH 575 Final Project

Christine Lind

Department of Applied Mathematics
University of Washington

June 2, 2005
Outline

Introduction
  System Description
  Exact System
  High Bounce Approximation

Experiments & Simulations
  Speaker Experiment
  Bouncing Ball Program
  Matlab Simulations

Comparisons
  “Standard” Map
  Bifurcation
  Strange Attractor

Conclusions

Extras
Simple Physical System

Interaction between:

- Ball
- Sinusoidally Oscillating Table
Bouncing Ball System Description

Initial Assumptions:

- $(x_k, t_k)$ - ball position and time of $k^{th}$ impact
  - Between Impacts, ball obeys Newton’s Laws:
    \[ x(t) = x_k + v_k(t - t_k) - \frac{g}{2}(t - t_k)^2 \quad \text{for} \quad t_k \leq t \leq t_{k+1} \]
  - Table is unaffected by impacts:
    \[ s(t) = A(\sin(\omega t + \theta_0) + 1) \]
**Bouncing Ball System Description**

---

**Solve for Next Impact Time**

\[ d(t) = x(t) - s(t) \] - distance between ball and table

- First \( t > t_k \) where \( d(t) = 0 \) is \( t_{k+1} \), next impact time!

\[
0 = d(t_{k+1}) = x_k + v_k(t_{k+1} - t_k) - \frac{g}{2}(t_{k+1} - t_k)^2 \\
- A(\sin(\omega t_{k+1} + \theta_0) + 1)
\]

- Note that at time \( t_k \):

\[
x_k = s(t_k) = A(\sin(\omega t_k + \theta_0) + 1)
\]
Bouncing Ball System Description

Solve for Next Impact Time

\[ d(t) = x(t) - s(t) \] - distance between ball and table

- Then the (Implicit) Time-Equation is:

\[ 0 = A \sin(\omega t_k + \theta_0) + v_k(t_{k+1} - t_k) - \frac{g}{2}(t_{k+1} - t_k)^2 \]

- Note that \( v_k \) is still unknown

- Find Velocity-Equation
Bouncing Ball System Description

Solve for Impact Velocity

First look at two different frames of reference

(a) Ground (Lab) Frame of Reference:
   ▶ $v_k$ - ball velocity at impact $k$
   ▶ $u_k$ - table velocity at impact $k$

(b) Table Frame of Reference:
   ▶ $\vec{v}_k = v_k - u_k$ - ball velocity at impact $k$
Bouncing Ball System Description

Solve for Impact Velocity

\[ \overline{v}'_k \] - velocity just before impact \( k \)
\[ \overline{v}_k \] - velocity just after impact \( k \)
\( \alpha \) - coefficient of restitution (describes damping)

- \( \overline{v}_k = -\alpha \overline{v}'_k \)
- \( 0 \leq \alpha \leq 1 \)
  - \( \alpha = 1 \) - no energy loss (no damping - elastic collision)
  - Transform back to Ground (Lab) Reference Frame...

\[ \overline{v}_k = v_k - u_k \]
Bouncing Ball System Description

Bouncing Ball System Description

<table>
<thead>
<tr>
<th>Solve for Impact Velocity</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_{k+1} = (1 + \alpha)u_{k+1} - \alpha v'_{k+1}$</td>
</tr>
</tbody>
</table>

- Recall that for $t_k \leq t \leq t_{k+1}$ the ball position is described by:

  \[
x(t) = x_k + v_k(t - t_k) - \frac{g}{2}(t - t_k)^2
  \]

  \[
v'_{k+1} = x'(t_{k+1}) = v_k - g(t_{k+1} - t_k)
  \]

- The table position is given by:

  \[
s(t) = A (\sin(\omega t + \theta_0) + 1)
  \]

  \[
u_{k+1} = s(t_{k+1}) = A\omega \cos(\omega t_{k+1} + \theta_0)
  \]

- Then we can solve for the Impact Velocity
Solve for Impact Velocity

Impact Velocity Equation:

\[ v_{k+1} = (1 + \alpha)A\omega \cos(\omega t_{k+1} + \theta_0) - \alpha (v_k - g(t_{k+1} - t_k)) \]
Bouncing Ball Exact Equations

System is described by

- **Time Equation:**
  
  \[
  0 = A \sin(\omega t_k + \theta_0) + v_k(t_{k+1} - t_k) \\
  - \frac{g}{2} (t_{k+1} - t_k)^2 - A \sin(\omega t_{k+1} + \theta_0)
  \]

- **Velocity Equation:**
  
  \[
  v_{k+1} = (1 + \alpha)A \omega \cos(\omega t_{k+1} + \theta_0) \\
  - \alpha (v_k - g(t_{k+1} - t_k))
  \]
Non-Dimensionalization!

Too many parameters to study the system efficiently
Parameters - $\alpha, A, \omega, g$

- Transform system into dimensionless variables:

\[
\begin{align*}
\theta_k &= \omega t_k + \theta_0 \\
\nu_k &= \frac{2\omega}{g} \nu_k
\end{align*}
\]

- New Parameter

\[
\beta = \frac{2\omega^2(1 + \alpha)A}{g}
\]
Bouncing Ball Exact Equations

Dimensionless System is described by

- Phase Equation:

\[
0 = \beta (\sin \theta_k - \sin \theta_{k+1}) + (1 + \alpha) (\nu_k (\theta_{k+1} - \theta_k) - (\theta_{k+1} - \theta_k)^2)
\]

- Velocity Equation:

\[
\nu_{k+1} = \beta \cos \theta_{k+1} - \alpha (\nu_k - 2(\theta_{k+1} - \theta_k))
\]

- Now we can study the system simply by varying \(\alpha\) and \(\beta\).
# Bouncing Ball Exact Equations

**Dimensionless System is described by**

- **Phase Equation:**
  
  \[
  0 = \beta (\sin \theta_k - \sin \theta_{k+1}) + (1 + \alpha) (\nu_k (\theta_{k+1} - \theta_k) - (\theta_{k+1} - \theta_k)^2)
  \]

- **Velocity Equation:**
  
  \[
  \nu_{k+1} = \beta \cos \theta_{k+1} - \alpha (\nu_k - 2(\theta_{k+1} - \theta_k))
  \]

- **Implicit Maps can be hard to analyze & simulate \(\Rightarrow\) make an approximation that will give us an Explicit Map...**
High Bounce Approximation

Assume:
change in table height $\ll$ maximum height of the ball

- Ball orbit symmetric about the maximum height:

$$x_k = x_{k+1} \quad v'_{k+1} = -v_k$$
High Bounce Approximation

\[ v'_{k+1} = -v_k \]

- Recall:

\[ v'_{k+1} = v_k - g(t_{k+1} - t_k) = -v_k \]

- Explicit Time Map:

\[ t_{k+1} = t_k + \frac{2}{g}v_k \]

- Use equation above to solve for the velocity map, and non-dimensionalize...
Bouncing Ball Approximation

### High Bounce Equations

- **Phase Equation:**
  \[ \theta_{k+1} = \theta_k + \nu_k \quad \text{(mod } 2\pi) \]

- **Velocity Equation:**
  \[ \nu_{k+1} = \alpha \nu_k + \beta \cos(\theta_k + \nu_k) \]

- For \( \alpha = 1 \), this is the “Standard” Map!
Outline

Introduction
  System Description
  Exact System
  High Bounce Approximation

Experiments & Simulations
  Speaker Experiment
  Bouncing Ball Program
  Matlab Simulations

Comparisons
  “Standard” Map
  Bifurcation
  Strange Attractor

Conclusions

Extras
Speakers and Function Generators

The physical system can be explored using a setup similar to the schematic shown below.
Experimental Set-up

Nicholas B. Tufillaro’s experimental set-up at Bryn Mawr College (circa 1985).
Bouncing Ball Experiment

Experimental Results
Nicholas B. Tufillaro’s experimental results, (see references)
Nicholas B. Tufillaro wrote a program called Bouncing Ball for the Apple Macintosh. Bouncing Ball simulates experiments by numerically solving the exact equations for the system. (You can download this program from his website)

- Bisection Method used (NOT Newton’s Method) because of ease of coding and stability
Bouncing Ball Simulations

default settings show four different windows:

- Trajectory
- Impact Data
- Animation
- Impact Map
Bouncing Ball Program

Bouncing Ball Simulations

Program can also plot

▶ Bifurcation Diagrams
▶ Basins of Attraction - for periodic points of period 1, 2, 3, 4, 8
▶ ...and play sounds at impact events - hear chaos!
It is easy to iterate the High Bounce Approximation, or “Standard” Map in Matlab. Initial conditions propagated in the following Matlab figures are shown below:
Matlab Simulations - High Bounce Approximation

Recall that the map in question is given by:

\[ \theta_{k+1} = \theta_k + \nu_k \pmod{2\pi} \]

\[ \nu_{k+1} = \alpha \nu_k + \beta \cos(\theta_k + \nu_k) \]

This map has fixed points \((\theta, \nu)\):

1. \(\left( \pm \arccos \left( \frac{2k\pi(1-\alpha)}{\beta} \right), 2k\pi \right)\)
2. for integer values of \(k\)
Matlab Simulations - High Bounce Approximation

For $\alpha = 1$ we get exactly the “Standard” Map, and

$$\theta_{k+1} = \theta_k + \nu_k \quad \text{(mod } 2\pi)$$

$$\nu_{k+1} = \nu_k + \beta \cos(\theta_k + \nu_k)$$

fixed points $(\theta, \nu)$:

- $(\frac{\pi}{2}, 2k\pi)$  \quad \left(\frac{3\pi}{2}, 2k\pi\right)$

- for integer values of $k$
High Bounce Approximation - $\alpha = 1, \beta = 1$

(This is the Standard Map)
Bouncing Ball Matlab Simulations

High Bounce Approximation

What will happen as we turn on the dissipation in the system?

- Center at \( \left( \frac{\pi}{2}, 0 \right) \) becomes Stable
- Centers at \( \left( \frac{\pi}{2}, 2k\pi \right) \) shift and become Stable (for \( \alpha \) not “too small”)
High Bounce Approximation - $\alpha = .999$, $\beta = 1$
High Bounce Approximation - $\alpha = .99$, $\beta = 1$
High Bounce Approximation - $\alpha = .9, \beta = 1$
Introduction
  System Description
  Exact System
  High Bounce Approximation

Experiments & Simulations
  Speaker Experiment
  Bouncing Ball Program
  Matlab Simulations

Comparisons
  “Standard” Map
  Bifurcation
  Strange Attractor

Conclusions

Extras
High Bounce - Exact System Comparison

Visually compare Standard Map $\alpha = 1$, $\beta = 1$ to Bouncing Ball Simulation with

- $f = 60$ Hz $\quad A = 0.00172563$ $\quad \alpha = 1$
- $\beta = \frac{2\omega^2(1+\alpha)A}{g} = \frac{2(2\pi*60)^2(1+1)*0.00172563}{981} \approx 1$

Then compare the Matlab Simulation with $\alpha = 0.9$, $\beta = 1$ with Bouncing Ball and

- $f = 60$ Hz $\quad A = 0.00181645$ $\quad \alpha = 0.9$
- $\beta = \frac{2\omega^2(1+\alpha)A}{g} = \frac{2(2\pi*60)^2(1+0.9)*0.00181645}{981} \approx 1$
High Bounce - Exact System Comparison $\alpha = 1, \beta = 1$
High Bounce - Exact System Comparison $\alpha = 0.9, \beta = 1$
**Exact System Basin of Attraction** $\alpha = 0.9, \beta = 1$

blue points get stuck to the table

- High Bounce Approx. is invertible $\Rightarrow$ cannot capture this behavior
High Bounce - Exact System Comparison

High Bounce Approx. also cannot describe situations where the ball rests on the table, but eventually leaves the table again - “Sticking Solutions”
Bifurcation

Exact System exhibits the classic “Period-Doubling” route to chaos for $\alpha = 0.5$:
High Bounce Approx. reproduces “Period-Doubling” route to chaos for $\alpha = 0.5$.
Bifurcation

High Bounce Approx. reproduces “Period-Doubling” route to chaos for $\alpha = 0.5$
Strange Attractor

Both models are shown for $\alpha = 0.5$, $\beta = 5.5$
Outline

Introduction
  System Description
  Exact System
  High Bounce Approximation

Experiments & Simulations
  Speaker Experiment
  Bouncing Ball Program
  Matlab Simulations

Comparisons
  “Standard” Map
  Bifurcation
  Strange Attractor

Conclusions

Extras
Simple Physical system $\Rightarrow$ Chaos

Interaction between:

- Ball
- Sinusoidally Oscillating Table

Leads to chaotic behavior for certain parameter values

- Experimental set-up can allow physical measurement of Feigenbaum’s constant
High Bounce - Exact System Comparison

Models have good qualitative agreement overall

► Exact System
  ▶ Implicit Equations - Hard to Solve/Simulate
  ▶ Can describe “sticking solutions” (not invertible)

► High Bounce Approximation
  ▶ Explicit Equations - Easy to Solve/Simulate
  ▶ Invertible \( \Rightarrow \) cannot describe “sticking solutions”
  ▶ Can describe non-physical situations (ball below table)
Questions?
Bouncing Ball System - References


▶ N.B. Tufillaro’s Website (Contains more references)
http://www.drchaos.net/drchaos/bb.html
Bouncing Ball System - Extras

"Standard" Map with $\alpha=0.99$, $\beta=1$
"Standard" Map with $\alpha=0.9, \beta=1$
"Standard" Map with α=0.9, β=1
Bouncing Ball System - Extras
Bouncing Ball System - Extras

Bifurcation Diagram for $\phi_n$, $v_n$ for varying $\beta$, $\alpha=0.5$
"Standard Map" with $\alpha=0.5$, $\beta=4$
Bouncing Ball System - Extras

"Standard Map" with $\alpha=0.5, \beta=4.8$
*Standard Map* with $\alpha=0.5$, $\beta=5$
"Standard Map" with $\alpha=0.5$, $\beta=5.5$
Bouncing Ball System - Extras
Feigenbaum’s Delta

\[ \delta = \lim_{n \to \infty} \frac{\lambda_n - \lambda_{n-1}}{\lambda_{n+1} - \lambda_n} = 4.669202 \]

\( \lambda \) is the value of A for which bifurcation occurs:

- \( A_1 = \lambda_1 = 0.0106, \quad A_2 = \lambda_2 = 0.0115, \)
- \( A_3 = \lambda_3 = 0.0117 \)

\[ \delta \approx \frac{0.0115 - 0.0106}{0.0117 - 0.0115} = 4.5 \]