Pendulum Simulation 2: Changing Length or Gravity

Tom Van Baak, tvb@LeapSecond.com

Introduction

In the previous article I described a pendulum simulation computer program in detail [1]. To at least 12 digits of accuracy the simulator correctly measured pendulum period and circular error. These tests were simply to confirm that the simulator matched existing mathematics. Of course if a simulator can only confirm what we already know then it's not worth much.

In this article, the simulator is used to discover something that maybe we don't already know. Period, T, is a function of pendulum length, L, and the acceleration of gravity, g. We all know that $T \approx 2\pi\sqrt{L/g}$ and so if L or g varies, the period varies ± half as much. But what happens to amplitude and velocity? I modified the simulator to find out and the results are quite interesting.

Over the long-term (weeks to months) environmental or metallurgical variations in L erode good timekeeping. Over the short-term (hours to days) astronomical variations in g affect only the best pendulum clocks. Understanding exactly what happens when L or g change is the goal.

How to measure amplitude, velocity, and period

The quantities *amplitude* (peak angle from center), *velocity* (peak velocity at center swing), and *period* (or its inverse, *rate*) are three key parameters that we can measure in a pendulum clock. Period is the most important since timekeeping is the sum of successive periods. Amplitude is important because period is affected by unwanted changes in amplitude – due to circular error. In practice, making precise measurements of velocity is much easier than making precise measurements of amplitude. So velocity is important because it is used to "calculate" amplitude.

A number of modern pendulum clocks make velocity measurements, use that to infer amplitude, and then employ some sort of servo to dynamically adjust the amount of impulse. The result is a closed loop system where constant amplitude is maintained. This amplitude control reduces the effects of circular error, improves the stability (consistency) of period, and thus improves timekeeping. It seems like a clever idea.

Many of us realize that these clocks are not strictly <u>amplitude</u> controlled; they are in fact <u>velocity</u> controlled, because what is being directly measured is velocity, not amplitude. For that matter, since energy goes as v^2 these clocks perhaps could be called <u>energy</u> controlled. There is concern about how the velocity is actually measured, especially when a highly accurate quartz crystal is used to make the measurement. An unintended consequence may be that the stability of the quartz is actually contributing to the stability of the pendulum clock [2].

To make velocity measurements an optoelectronic gate is placed at the center of the swing and the short interruption of the beam gives two measurements for the price of one. First, a time measurement from gate to gate to gate gives period. Second, a time measurement of the short interval that the beam is interrupted gives a measurement of velocity. This technique works rather well.

Questions arise if one wants high precision. The wider the gate the more the measurement is a low *average* velocity rather than true *peak* velocity. The narrower the gate, the shorter the time interval and the less precise the measurement is. So there is an unwelcome but unavoidable compromise between accuracy and resolution. In addition, the calculation of amplitude is both indirect, because it is based on velocity, and inexact, because of assumptions made in the conversion formula. For all these reasons trying to understand subtle details about amplitude and velocity is difficult or maybe even misleading in a real pendulum clock.

That's enough about real pendulum clocks and measurement. Now back to simulation…

The simulation program maintains two variables, *theta* and *omega*. Theta is the instantaneous angle of the pendulum and omega is the instantaneous velocity. While the program is simulating forces, position, and motion – micro step by micro step – these two variables are checked for change in sign.

When omega changes from negative to positive the pendulum has reached the left edge. When theta changes from negative to positive the pendulum has just passed center, going right. When omega changes from positive to negative the pendulum has reached the right edge. And when theta changes from positive to negative the pendulum has just passed center again, going left.

By capturing the values of theta and omega at these points the simulator can make highly accurate "measurements" of peak amplitude and peak velocity, as well as period. Unlike a real pendulum clock, these measurements are all <u>direct</u>, <u>accurate</u>, and can be made as <u>precise</u> as needed. In particular, amplitude measurements are true amplitude, and not a velocity measurement disguised as amplitude.

How to vary gravity, g

Most discussions of pendulum performance assume some fixed value for g, the acceleration of gravity. In the case of the very best pendulum clocks, the fact that g varies gradually over time as a result of lunar/solar tides is of great interest. The question is – how exactly does a pendulum react when g slowly changes?

It may be too costly to buy, or too difficult to build, a pendulum clock that is precise enough to make amplitude, velocity, and period measurements while g slowly varies over the course of the day. The perturbation is on the order of 0.1 ppm and only a few clocks exist that are sensitive enough to "detect" this effect. But making a sensitive pendulum clock in software is easy. So instead of spending a lifetime to build the world's best pendulum clock, I spent a minute to add variations in g to the simulator.

In the old code, g was a fixed constant:

```
// Perform microstep integration (Euler-Cromer).
omega -= dt * sin(theta) * g / L;
theta += dt * omega;
```

In the new code, the fixed constant is g0 and g is now a variable with a small amplitude (g_amp) slow period (g_per) sinewave modulation:

```
g_rad = 2 * PI * dt / g_per;
g = g0 * (1 + g_amp * sin(steps * g_rad));
// Perform microstep integration (Euler-Cromer).
omega -= dt * sin(theta) * g / L;
theta += dt * omega;
```

It's that simple. Note there are many ways to "vary g": instantaneous, ramp, triangle, exponential, sinusoidal, etc. To mimic how gravity varies in real life, sinusoidal modulation of g was chosen. For the purpose of the plots below, the choice of modulation amplitude and period is somewhat arbitrary. A gravity variation of 1% and modulation period of 12 hours makes a nice illustration.

In this case – instead of g being constant 9.8 – g starts at 9.800 and gradually grows to 9.898 (1% higher than normal) over 3 hours, smoothly heads back to 9.800 by 6 hours, continues falling to a low of 9.702 (1% lower than normal) at 9 hours, and heads back to normal 9.800 after 12 hours, and so on. All this occurs as a smooth sine wave.

Simulation speed is 30% slower when g modulation is enabled because a new value of g is computed at every step inside the integration loop. But this way the simulated pendulum experiences exceptionally smooth changes in force, just like a real pendulum. And the program can still perform millions of steps per second.

How to vary length, L

Variations in L are interesting because this also happens in real live; for example, as a result of temperature or other environmental changes. The question is – how exactly does a pendulum react when L slowly changes?

At first glance one might think the code changes for L are equivalent to the changes for g. But variations in length are more complex than variations in gravity. This is because when length changes during a pendulum swing the equations of motion are affected. Imagine if the rod were a string. To reduce the length of the "rod" you pull the string. This pull on a swinging string effectively adds another force term. Besides the usual *acceleration of gravity* we now also have *acceleration of string* and this must be incorporated into the simulation equation to properly reflect the net force on the bob. This is a Coriolis term, according to a recent Am. J. Phys. article [3].

In the old code, L was a fixed constant:

```
// Perform microstep integration (Euler-Cromer).
omega -= dt * sin(theta) * g / L;
theta += dt * omega;
```

In the new code, the fixed constant is L0 and L is now a variable with small amplitude (L_amp) slow period (L_per) sinewave modulation. In addition the equation includes both the main force of gravity and the small, non-zero force of a slowly expanding/contracting rod:

```
L_rad = 2 * PI * dt / L_per;
dL = L0 *      L_amp * cos(steps * L_rad) * L_rad;
 L = L0 * (1 + L_amp * sin(steps * L_rad));
```

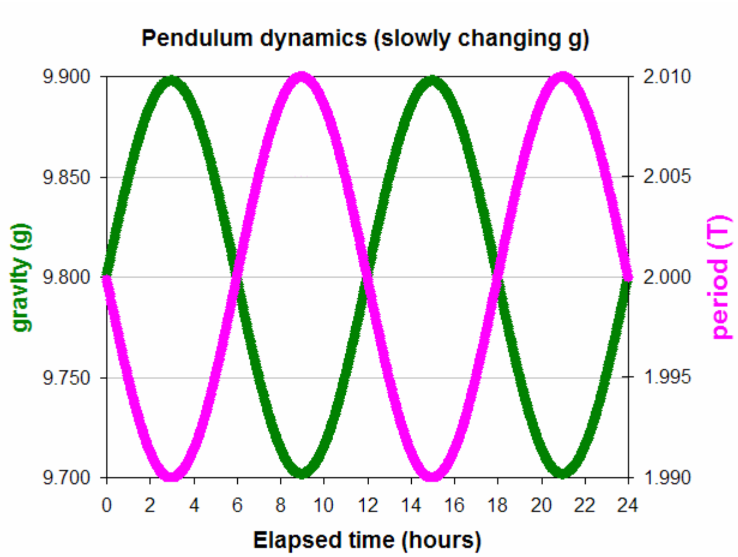
```
// Perform microstep integration (Euler-Cromer).
omega -= dt * sin(theta) * g / L + 2 * dL * omega / L;
theta += dt * omega;
```

With these two new features added, the simulation program is ready to tackle what happens to a pendulum when gravity and/or length changes.

What changes when gravity changes?

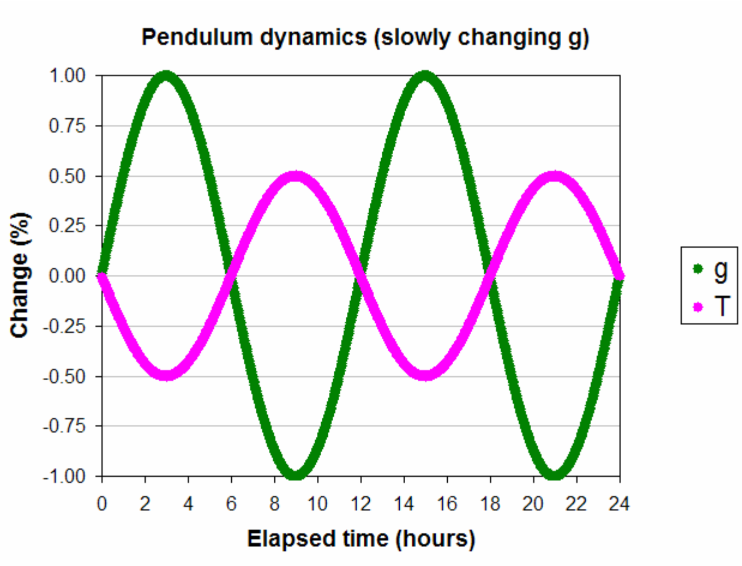
I simulated a standard ~1 meter, ~2 second period pendulum in normal gravity (~9.8 m/s²) and 1° amplitude. For each period, the simulator measures various quantities like amplitude, velocity (center swing), height (at end of swing), energy (sum of KE and PE anywhere along the path), and period.

For these plots, a gravity variation of 1% and modulation period of 12 hours was chosen. Below is a plot from the raw simulation data showing g (green) and T (pink) over the course of a simulated 24 hour day:



The plot shows that g varies from 9.7 to 9.9 m/s², which is ±1% of 9.8, as we specified. The simulation shows that T varies from 1.99 to 2.01 s, which is ±½% of 2.0, and opposite phase as g.

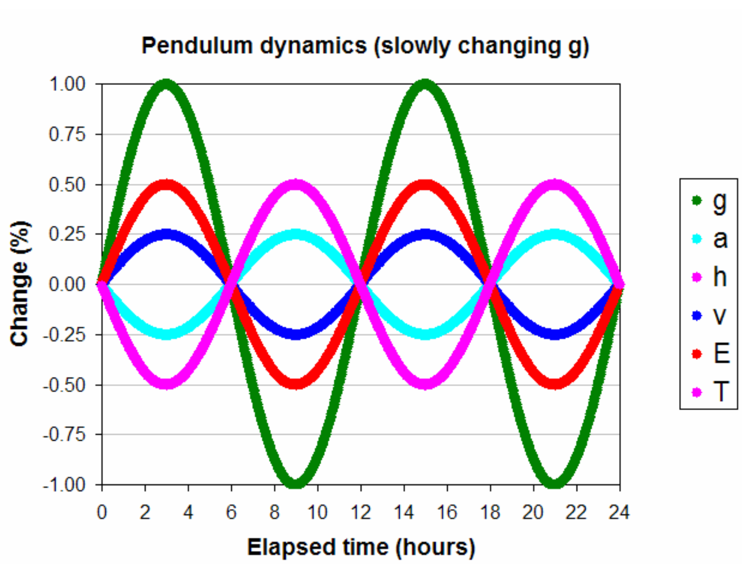
The problem with this plot is that it doesn't give a clear sense of proportion. Both curves appear the same size and the reader is forced to carefully look at the axis labels to determine the magnitude of the variations. An alternative is to use a relative (geometric) scale instead of an absolute (arithmetic) scale for the y-axis. Then the full-scale is ±1% for all measurements and the plot becomes:



This plot much better conveys the relationship between gravity and period.

Since $T \approx 2\pi\sqrt{(L/g)}$ this simulation plot is exactly what we would expect – that period changes half as much as gravity changes (due to the square root) and with opposite sign (due to g being in the denominator). That is, as gravity increases, then period decreases (and rate or frequency increases). Of course we don't need simulation to tell us this, but it's nice that simulation gives the correct result.

Next we add more measurements to the same plot: velocity (dark blue), energy (red), amplitude (light blue), and height (pink). By height I mean the elevation of the bob at end of swing. We see as gravity increases, then velocity increases at a quarter the rate, and energy increases at half the rate. This makes sense since energy is proportional to v^2. We also see that amplitude decreases as gravity increases.



Note that amplitude changes in the opposite direction as velocity. That is, as gravity increases, velocity increases but amplitude decreases.

This is exactly the opposite of the usual constant gravity case where as velocity increases so does amplitude. It is this amplitude ≈ velocity assumption that allows one to measure velocity and then treat it like it is an amplitude measurement. This assumption breaks in the case where gravity slowly changes. This might worry anyone who is using velocity measurement to accomplish amplitude control.

The simulation plot above was a relief for me because it agrees with my earlier results, derived from applying adiabatic invariance, a completely different approach [4].

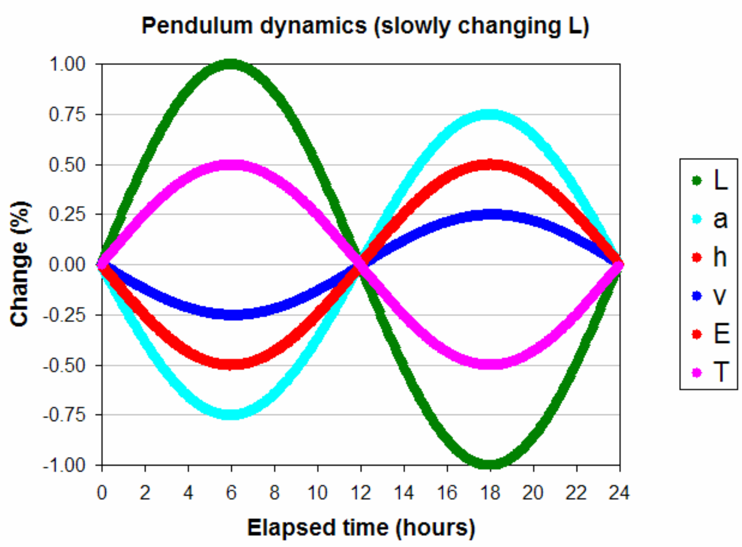
In summary, as the acceleration of gravity increases, pendulum PE and KE increases by ½ as much, velocity increases by ¼, period decreases by ½, height follows period, and amplitude decreases by ¼. Another way to show this is:

If $\Delta g/g = 1$, then $\Delta T/T = -\frac{1}{2}$, $\Delta E/E = +\frac{1}{2}$, $\Delta v/v = +\frac{1}{4}$, $\Delta h/h = -\frac{1}{2}$, $\Delta a/a = -\frac{1}{4}$.

So now we know what exactly happens when gravity changes.

What changes when length changes?

Here are the results of the length simulation. For these plots, a length variation of 1% and modulation period of 24 hours was chosen. Below is a plot from the raw simulation data showing L (green) and T (pink) over the course of a simulated 24 hour day:



There are many similarities with the g plots. Period, velocity, energy react as expected but amplitude is unusual. Is this a bug or is it real?

In summary, as the length of the rod increases, pendulum PE and KE decreases by ½ as much, velocity decreases by ¼, period increases by ½, height follows energy, and amplitude decreases by ¾. Another way to show this is:

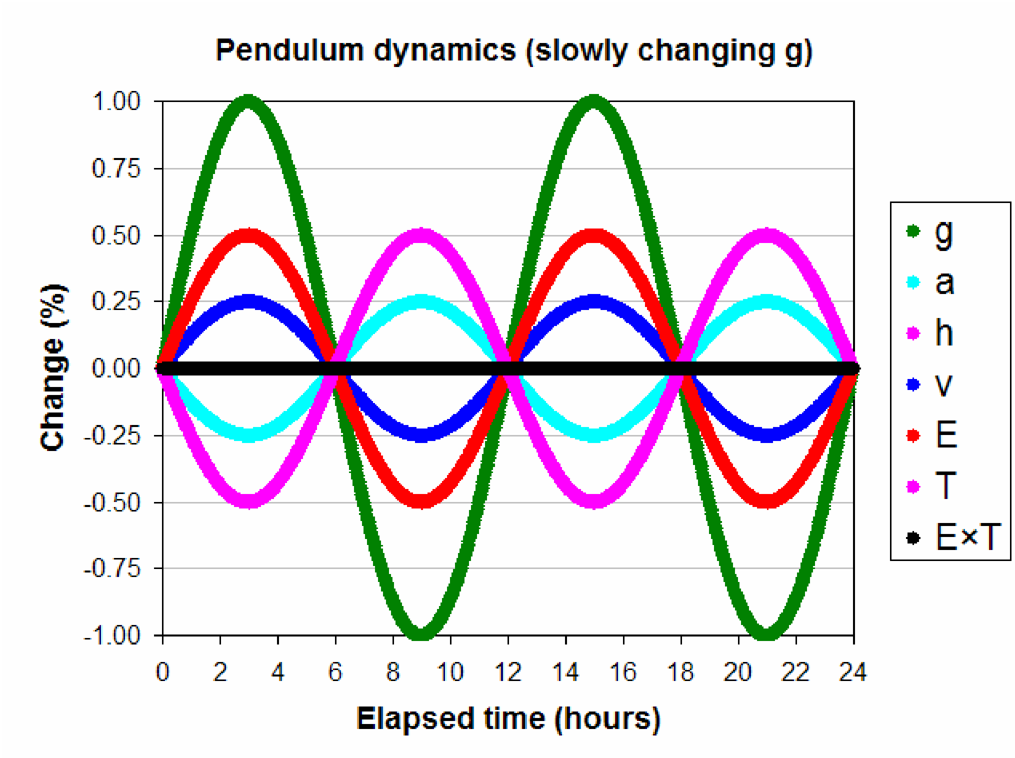
If $\Delta L/L = 1$, then $\Delta T/T = +\frac{1}{2}$, $\Delta E/E = -\frac{1}{2}$, $\Delta v/v = -\frac{1}{4}$, $\Delta h/h = -\frac{1}{2}$, $\Delta a/a = -\frac{3}{4}$.

So now we know what happens when length changes.

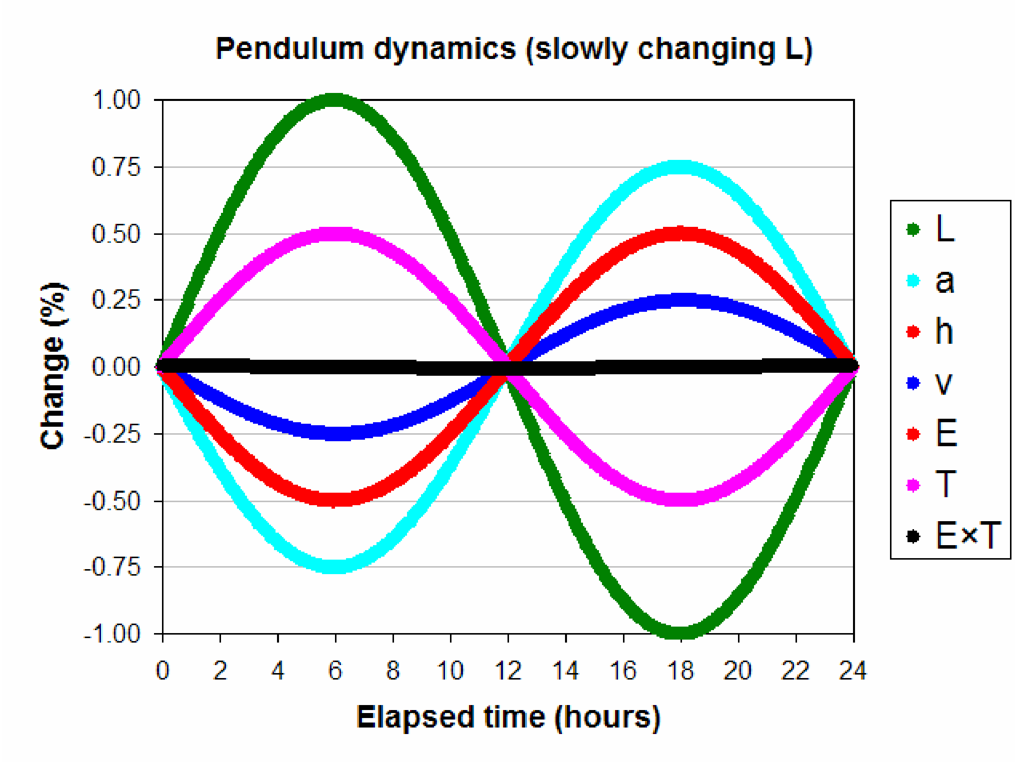
Adiabatic Invariants

I mentioned earlier that I had obtained the "varying g" results some time ago by applying the principle of adiabatic invariance, a topic from classical mechanics. Although I believed the answer was correct, several readers questioned my use of adiabatic invariance. In physics it's sometimes hard to believe conclusion Y if you have to first believe condition X.

In that theory, energy times period of a pendulum is alleged to be a constant – even if parameters like L and g slowly change all the time [5]. That means if we were to make a plot, even though *every other quantity is changing*, the quantity E×T should not change. That's quite a bold prediction. And, well, here is the g plot:



And the L plot:



We see that in both cases, the black line, E×T, is flat. So pendulum simulation scores another victory – it has confirmed that E times T is an adiabatic invariant, just like theory says. This is very satisfying. It's almost as if the entire universe is just a big fast simulator.

Another interesting observation – since E×T is constant then for best timekeeping (constant T) it seems the goal should be constant E, which is the same as constant v, regardless of amplitude. So from this perspective a constant velocity control loop is the ideal, even if it means that amplitude will vary. But then what about circular error? I sense deep thought and more simulation coming soon.

Conclusion

We now see how pendulum variables change as L or g change. I'm not sure this has any impact on normal pendulum clocks. However, for ultra high precision pendulum clocks, especially those which use optoelectronic sensing for "amplitude control", these results may be revealing.

We know that g changes by about 0.1 ppm in a quasi-periodic manner. We also know that the effective length of a pendulum rod can change periodically (primarily due to temperature) or gradually (due to drift) or sporadically (e.g., invar) so these results may have practical value.

This set of experiments with simulation is another step towards exploring how precision pendulum clocks operate. Although this is being done with an imaginary friction-less pendulum, a next step is to explore the role of friction, impulse, and amplitude control methods.

The C code (and Windows EXE) for the new version of pendulum simulator is available. [6]

The simulation uncovered interesting relationships among amplitude and velocity in the presence of changing L and g. In particular, we see that when gravity or length changes, amplitude and velocity do not follow each other.

Here is a summary of the results:

If $\Delta g/g = 1$, then $\Delta T/T = -\frac{1}{2}$, $\Delta E/E = +\frac{1}{2}$, $\Delta v/v = +\frac{1}{4}$, $\Delta h/h = -\frac{1}{2}$, $\Delta a/a = -\frac{1}{4}$.

If $\Delta L/L = 1$, then $\Delta T/T = +\frac{1}{2}$, $\Delta E/E = -\frac{1}{2}$, $\Delta v/v = -\frac{1}{4}$, $\Delta h/h = -\frac{1}{2}$, $\Delta a/a = -\frac{3}{4}$

Notes

[1] Pendulum Simulation 1: Period and Circular Error, Horological Science Newsletter, 2014-1

[2] Understanding Hybrid Quartz / Pendulum Clocks, Horological Science Newsletter, 2012-1

[3] Unraveling a classical mechanics brain twister, Norman Paris, Michael L. Broide, Am. J. Phys. 79, 1250 (2011), dx.doi.org/10.1119/1.3636655

[4] What Changes When Gravity Changes (Tides, part 4), Horological Science Newsletter, 2009-5

[5] http://en.wikipedia.org/wiki/Adiabatic_invariant

[6] www.leapsecond.com/tools/pend11.exe (pend11.c) is a simulator to vary L and g

[7] Copies of all my HSN pendulum papers: http://leapsecond.com/hsn2006/