[a]Electronic mail: andrejj@fiz.uni-lj.si
[1]Gregory W. Swift, ''Thermoacoustic engines,'' J. Acoust. Soc. Am. **84**, 1145–1180 (1988).
[2]J. Wheatley, T. Hofler, G. W. Swift, and A. Migliori, ''Understanding some simple phenomena in thermoacoustics with applications to acoustical heat engines,'' Am. J. Phys. **53**, 147–162 (1985).
[3]Gregory W. Swift, ''Thermoacoustic engines and refrigerators,'' Phys. Today **22–28**, (1995).

Listening to the coefficient of restitution and the gravitational acceleration of a bouncing ball

C. E. Aguiar[a)] and F. Laudares[b)]
Instituto de Física, Universidade Federal do Rio de Janeiro, Cx.P. 68528, Rio de Janeiro, 21945-970, RJ, Brazil

(Received 15 July 2002; accepted 4 October 2002)

We show that a well known method for measuring the coefficient of restitution of a bouncing ball can also be used to obtain the gravitational acceleration.
[DOI: 10.1119/1.1524166]

Three contributions to this journal have described how to measure the coefficient of restitution between a ball and a flat surface using the sound made by the collision of the ball with the surface.[1–3] The procedure reported in these papers is to drop the ball vertically on a horizontal surface, allow it to bounce several times, while recording the sound produced by the impacts. Analysis of the recording gives the time intervals between successive rebounds, and from these the coefficient of restitution is obtained.

The evolution of the techniques described in these papers is a nice example of how the development of microcomputers has changed the science teaching laboratory. In 1977, Bernstein[1] detected the sound with a microphone, amplified and filtered the signal, and fed it to a pen recorder. Smith, Spencer, and Jones,[2] in 1981, connected the microphone to a microcomputer via a homemade data collection and interface circuit, and then uploaded the resulting data to a larger computer for analysis and graphical display. In 2001, Stensgaard and Lægsgaard[3] used the microphone input of a PC sound card to make the recording, thus reducing the equipment requirements to a standard microcomputer.

To see how the coefficient of restitution ϵ is related to the time between bounces, note that if ϵ is constant (independent of velocity), and air resistance is negligible, the velocity of the ball just after the nth bounce on the fixed surface is given by

$$v_n = v_0 \epsilon^n, \tag{1}$$

where v_0 is the velocity just before the first impact. The time-of-flight T_n between the nth and $(n+1)$th collisions is proportional to v_n,

$$T_n = \frac{2v_n}{g}, \quad n = 1,2,\ldots\,, \tag{2}$$

where g is the gravitational acceleration. Thus

$$T_n = T_0 \epsilon^n, \tag{3}$$

where we have defined $T_0 \equiv 2v_0/g$. Taking the logarithm of both sides of Eq. (3) we obtain

$$\log T_n = n \log \epsilon + \log T_0, \tag{4}$$

so that the plot of $\log T_n$ vs n is a straight line of slope $\log \epsilon$ and intercept $\log T_0$. Thus, as long as it is independent of velocity, the coefficient of restitution can be obtained by fitting the straight line of Eq. (4) to the time-of-flight data.

The purpose of this note is to point out that this straight-line fit can also be used to determine another physical quantity of interest: the gravitational acceleration g. The (rather simple) observation is that, if the ball is released from a known height h, then $T_0 = (8h/g)^{1/2}$, and

$$g = \frac{8h}{T_0^2}. \tag{5}$$

Thus, just as the slope parameter of Eq. (4) fixes the coefficient of restitution, the intercept parameter determines the acceleration of gravity (if the easily measured initial height h is known).

In order to check how this works in practice, we have dropped a ''superball'' from a measured height onto a smooth stone surface and recorded the sound produced by the successive impacts. The recording was made with the microphone and sound card of a personal computer running Windows, using the sound recorder program that comes with the operating system. The sampling frequency was 22 050 Hz, resulting in a time resolution of 45 μs. The audio file, stored in the binary WAV format, was converted to ASCII text format with the shareware program AWAVE AUDIO.[4] The recorded signal is plotted in Fig. 1, where the pulses corre-

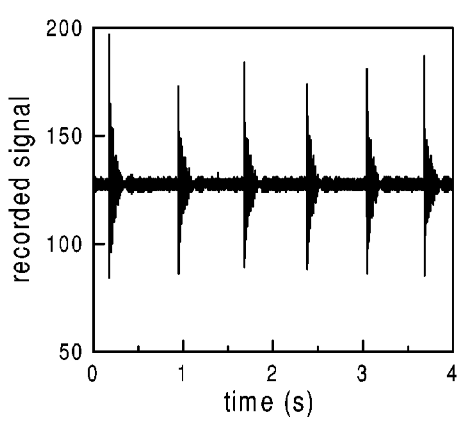

Fig. 1. The sound of a ball bouncing on a horizontal surface. The zero sound level corresponds to 128 on the vertical axis.

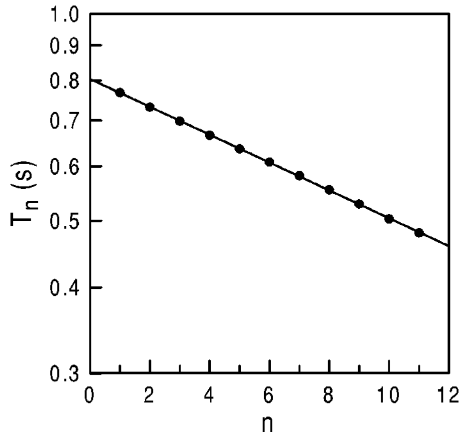

Fig. 2. Time-of-flight T_n between impacts n and $n+1$. The line is the least-squares fit using Eq. (4).

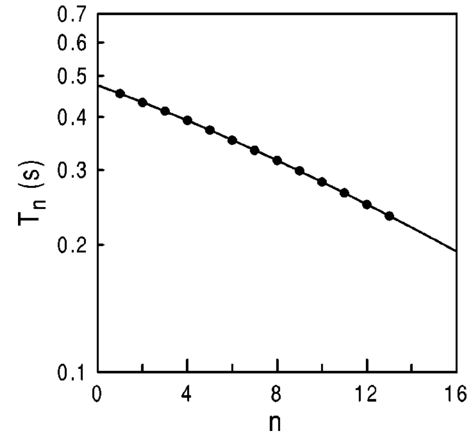

Fig. 4. Time-of-flight T_n between impacts n and $n+1$. The curve is the least-squares fit using Eq. (8).

sponding to individual impacts are easily recognized (only the first six collisions are shown). We have used 8-bit resolution in the recording, so that data values can only go from 0 to 255. The no-signal value corresponds to 128.

The time intervals T_n between collisions n and $n+1$ were obtained directly through inspection of the ASCII sound file. They are plotted in Fig. 2 (in logarithmic scale) as a function of n. The least-squares fit of the T_n data set to Eq. (4) gives

$$\epsilon = 0.9544 \pm 0.0002, \tag{6a}$$

$$T_0 = 0.804 \pm 0.001 \text{ s.} \tag{6b}$$

The best-fit line is also shown in Fig. 2.

The ball was released from a height $h = 79.4 \pm 0.1$ cm above the surface. Taking this and the adjusted T_0 into Eq. (5), we obtain for the gravitational acceleration

$$g = 982 \pm 3 \text{ cm/s}^2.$$

For comparison, the value of g in Rio de Janeiro is 978.8 cm/s^2.

The applicability of the method described above depends on ϵ being constant over the range of impact velocities involved in the experiment. That this condition is satisfied in the present case is seen in Fig. 3, where the coefficient of restitution for an impact at velocity v_n, $\epsilon = v_{n+1}/v_n = T_{n+1}/T_n$, is plotted as a function of T_n [recall that $v_n \propto T_n$, see Eq. (2)]. The coefficients of restitution for the different impacts are all very close to the least-squares value given in Eq. (6), indicated by the dashed line in Fig. 3.

A case in which the coefficient of restitution depends on the velocity is shown in Fig. 4, where we display the times of flight of a superball dropped from $h = 27.5 \pm 0.1$ cm onto a wood surface. A plot of ϵ at each collision, shown in Fig. 5, reveals a clear dependence of the coefficient of restitution on the time-of-flight (or impact velocity). Assuming a linear relation between ϵ and T, as suggested by Fig. 5,

$$\epsilon = \epsilon_0 (1 + \alpha T), \tag{7}$$

we obtain an extension of Eq. (3),

$$T_n = T_0 \epsilon_0^n \prod_{i=0}^{n-1} (1 + \alpha T_i). \tag{8}$$

The least-squares fit of Eq. (8) to the data shown in Fig. 4 gives

$$\epsilon_0 = 0.921 \pm 0.001, \tag{9a}$$

$$\alpha = 0.078 \pm 0.003 \text{ s}^{-1}, \tag{9b}$$

$$T_0 = 0.4752 \pm 0.0005 \text{ s.} \tag{9c}$$

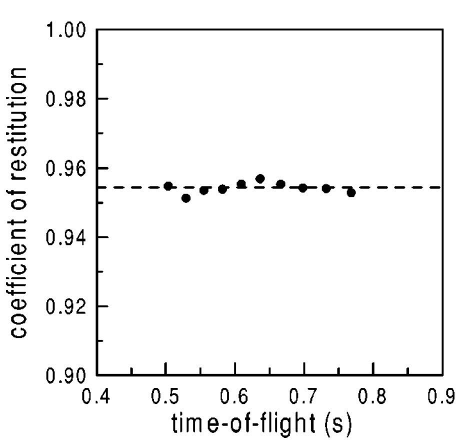

Fig. 3. The coefficient of restitution $\epsilon = T_{n+1}/T_n$ as a function of the time-of-flight T_n, for the data of Fig. 2. The dashed line indicates the adjusted value given in Eq. (6).

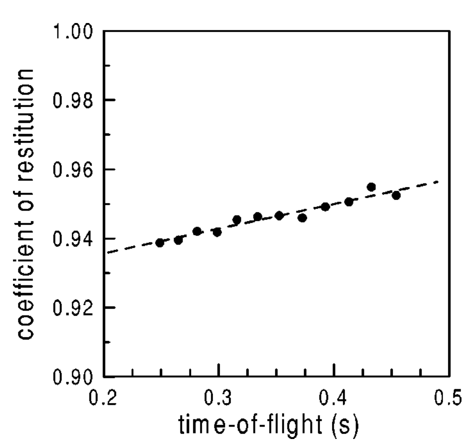

Fig. 5. The coefficient of restitution $\epsilon = T_{n+1}/T_n$ as a function of T_n, for the data of Fig. 4. The dashed line is the linear relation of Eq. (7) with the adjusted parameters given in Eq. (9).

The curves corresponding to these parameters are also shown in Figs. 4 and 5. The above value for T_0 yields

$$g = 974 \pm 5 \text{ cm/s}^2,$$

again a very reasonable value. Consideration of the velocity dependence of the coefficient of restitution was important in order to get an accurate result; had we assumed a constant ϵ, we would obtain $g = 935 \pm 10$ cm/s^2.

To summarize, we have seen that the value of the gravitational acceleration is a useful by-product of experiments devised to ''hear'' the coefficient of restitution of a bouncing ball. The measurement of g is particularly simple if the coefficient of restitution is independent of the impact velocity, but more complicated cases can also be handled.

After this work was completed we learned of a recent paper by Cavalcante *et al.*,[5] in which g was measured using the sound of a bouncing ball. The analysis presented in the paper is, however, somewhat different from ours. Another related reference is the paper by Guercio and Zanetti[6] in this journal.

[a]Electronic mail: carlos@if.ufrj.br
[b]Electronic mail: f_laudares@hotmail.com

[1]A.D. Bernstein, ''Listening to the coefficient of restitution,'' Am. J. Phys. **45**, 41–44 (1977).
[2]P.A. Smith, C.D. Spencer, and D.E. Jones, ''Microcomputer listens to the coefficient of restitution,'' Am. J. Phys. **49**, 136–140 (1981).
[3]I. Stensgaard and E. Lægsgaard, ''Listening to the coefficient of restitution—revisited,'' Am. J. Phys. **69**, 301–305 (2001).
[4]AwaveAudio (audio file format converter), FMJ-Software, http://www.fmjsoft.com
[5]M.A. Cavalcante, E. Silva, R. Prado, and R. Haag, ''O estudo de colisões através do som,'' Revista Brasileira de Ensino de Física **24**, 150–157 (2002).
[6]G. Guercio and V. Zanetti, ''Determination of gravitational acceleration using a rubber ball,'' Am. J. Phys. **55**, 59–63 (1987).

Automation of the Franck–Hertz experiment and the Tel-X-Ometer x-ray machine using LABVIEW

W. Fedak,[a] D. Bord, C. Smith, D. Gawrych, and K. Lindeman
Department of Natural Sciences, University of Michigan–Dearborn, Dearborn, Michigan 48128

(Received 10 June 2002; accepted 17 October 2002)

We describe the use of LABVIEW to automate data collection and instrument control for the Franck–Hertz experiment and for the popular Tel-X-Ometer x-ray machine. Such automation permits the rapid collection and reduction of large amounts of data, thus facilitating exploration of the basic physics of these experiments. The use of industry-standard software packages, such as ORIGIN and MATHEMATICA, provides students with valuable exposure to professional tools for the display and analysis of data.
[DOI: 10.1119/1.1527949]

I. INTRODUCTION

In this paper we describe the use of LABVIEW,[1] an industry-standard software package for data acquisition and instrument control, to automate the Franck–Hertz (FH) experiment and the Tel-X-Ometer[2,3] x-ray machine. The FH experiment is a standard advanced undergraduate laboratory activity that demonstrates the existence of quantized energy levels in atoms. The Tel-X-Ometer x-ray machine is a popular laboratory device for investigating a wide variety of phenomena associated with the production, absorption, and scattering of x-rays.[4,5] The automation of these experiments can reduce significantly the amount of time spent by a student performing routine data collection, as well as provide digitized formats that lend themselves to easy display, analysis, and comparison of data. Together, these enhancements of the experiments enable students to focus more directly on the physics of the investigation.

LABVIEW facilitates the development of graphical programs appropriately called *virtual instruments* (VIs). The user interface of a LABVIEW program resembles the front panel of a real laboratory instrument with dials, push buttons, toggles, and various digital and analog readouts. The program logic is created graphically by building block diagrams that consist of objects (functions and controls) and wires that transfer data between objects.

LABVIEW instrument drivers are available for hundreds of different data acquisition devices, allowing wide use of existing laboratory equipment.[6] LABVIEW versions are available for Microsoft Windows, Apple Macintosh, Sun Solaris, HP-UX, and Linux operating systems.[7] Given the increasingly common use of LABVIEW in college and university electronics courses, many students may already be familiar with the software.[8] For example, our instrumentation course, which is a prerequisite for the advanced lab course, now includes basic LABVIEW programming.

We have written LABVIEW VIs for both the FH experiment and for the Tel-X-Ometer x-ray machine. The FH VI provides fast and accurate data collection, as well as real-time display of the FH curves. The Tel-X-Ometer VI provides precise instrument control and automated data collection. The acquired data in both cases can be saved to a text-delimited file, permitting detailed data analysis and display using software packages such as ORIGIN[9] and MATHEMATICA.[10] Although LABVIEW routines can be written to analyze the acquired data and calculate various results (for example, the mean FH peak separation or the Cu K_α wavelength), we intentionally left the data analysis portion out of our LABVIEW VIs to avoid making each experiment a ''black box.'' Data analysis software such as ORIGIN and MATHEMATICA is ideally suited to such analysis and provides valuable ''real-world'' experience using professional software tools. Students with LABVIEW programming expertise are

