An acoustical analogue of a galactic-scale gravitational-wave detector

Michael T. Lam*

Department of Physics and Astronomy, West Virginia University, P.O. Box 6315, Morgantown, WV 26506, USA and Center for Gravitational Waves & Cosmology, West Virginia University, Chestnut Ridge Research Building, Morgantown, WV 26505, USA

Joseph D. Romano†

Department of Physics and Astronomy, University of Texas Rio Grande Valley, Brownsville, TX 78520, USA and Department of Physics and Astronomy, Texas Tech University, Box 41051, Lubbock, TX 79409, USA

Joey S. Key‡

Division of Physical Sciences, University of Washington Bothell, Bothell, WA 98011, USA

Marc Normandin§

Department of Physics and Astronomy, University of Texas Rio Grande Valley, Brownsville, TX 78520, USA

Jeffrey S. Hazboun¶

Center for Advanced Radio Astronomy, University of Texas Rio Grande Valley, Brownsville, TX 78520, USA

(Dated: June 22, 2018)

By precisely monitoring the "ticks" of Nature's most precise clocks (millisecond pulsars), scientists are trying to detect the "ripples in spacetime" (gravitational waves) produced by the inspirals of supermassive black holes in the centers of distant merging galaxies. Here we describe a relatively simple demonstration that uses two metronomes and a microphone to illustrate several techniques used by pulsar astronomers to search for and detect gravitational waves. An adapted version of this demonstration could be used as an instructional laboratory investigation at the undergraduate level.

I. INTRODUCTION

A *pulsar timing array* is a galactic-scale gravitational-wave detector, which can be used to search for gravitational waves from the inspiral of supermassive black-hole binaries (of order 10^9 solar masses) in the centers of distant galaxies[1–3]. The array consists of a set of galactic millisecond *pulsars*—rapidly-rotating neutron stars, which have masses of order the mass of the Sun and magnetic fields of order a billion times stronger than that of the Earth[4]. Millisecond pulsars rotate

nearly a thousand times each second (faster than a kitchen blender), emitting a narrow beam of radio waves along the separately-aligned magnetic axes that sweeps across the sky like a revolving beacon on top of a lighthouse. If this radio beam crosses our line of sight to the pulsar, a radio telescope on Earth will observe pulses of radiation, which arrive with a regularity that rivals (or even exceeds) that of the best atomic clocks[5].

By precisely monitoring the pulse arrival times over many years, radio astronomers can determine what the rotation period of the pulsar is, how the rotation is slowing down, whether the pulsar is orbiting a companion star, as well as how the interstellar medium affects the propagation of the pulses[4]. The difference between the *measured* times of arrival and the *expected* times of arrival (taking all of these effects into account) are called *timing residuals*. If the pulsar timing model is good, the residuals should be randomly scattered around zero with a root-mean-square (rms) amplitude determined by measurement noise in the radio receiver and statistical fluctuations in the pulses themselves. The residuals for an individual pulsar may be correlated in time[6–9] (so-called red noise), but residuals associated with different Earth-pulsar baselines should not be correlated with one another in the absence of any common external influence. Deviations from this expected behavior could be due to either an incomplete timing model (e.g., not realizing that the pulsar is in a binary) or the presence of low-frequency (nanohertz to microhertz) gravitational waves[10].

A gravitational wave passing between the Earth and a pulsar will stretch and squeeze space transverse to its motion, slightly advancing or retarding the arrival times of the individual pulses[1] (the original framework was in the context of spacecraft doppler tracking[11]). Unlike the measurement noise or intrinsic pulsar timing noise discussed above, the modulation of pulse arrival times induced by a gravitational wave will be *correlated* across different pulsars in the array, due to its common influence in the vicinity of the Earth. Moreover, this correlation will have a very specific dependence on the angle between a pair of Earth-pulsar baselines, the so-called *Hellings and Downs curve*[2] shown in Figure 1. The detection of this expected correlation in the timing residuals from an array of pulsars would be evidence for the presence of gravitational waves, similar to the recent detections by the advanced LIGO and Virgo detectors[12–14].

A. Metronomes and microphones

In order to illustrate how gravitational-wave astronomers are using correlation methods to search for gravitational waves, we have developed a demonstration using metronomes and a microphone, which serves as an *acoustical analogue* of a pulsar timing array. In this demonstration, radio

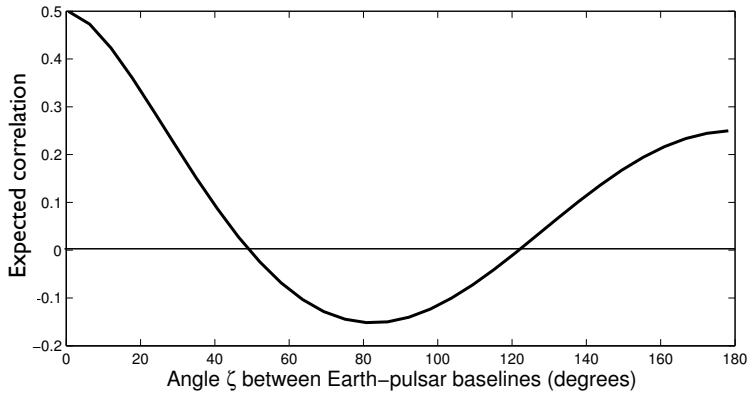

FIG. 1: Expected correlation coefficient between the timing residuals for a pair of Earth-pulsar baselines separated by angle ζ.

pulses from an array of Galactic pulsars are represented by ticks of an array of metronomes (only two metronomes are needed for this demonstration); radio receivers on Earth are represented by a single microphone; and the passage of a gravitational wave is represented by the motion of the microphone around its nominal position. The analogy is not perfect as the motion of the microphone does not represent a wave of any kind, and the correlations that it induces have a different angular dependence than that induced by a real gravitational wave[15]. But what is important is that there *are* correlations, as the microphone motion modulates the arrival times of the metronome pulses by changing the distance between the metronomes and the microphone. And although the angular dependence of the correlations for the microphone motion is different than that for gravitational waves, it is, nonetheless, a specific function of the angle between a pair of microphone-metronome baselines, which can be calculated theoretically and also verified experimentally by doing the demonstration.

In the following sections, we will describe the metronome-microphone demonstration in detail. In Section II, we describe the specific hardware (i.e., metronomes and microphone) and software routines that we use to do the analysis. In Section III, we list the techniques used in real pulsar timing analyses that are illustrated by the demonstration. Understanding these time-series analysis techniques for broader application, as well as a deeper understanding of using a pulsar timing array as a gravitational-wave detector, can be thought of as the *learning outcomes* for the demonstration. In Section IV, we discuss the two main parts of the demonstration (the single-metronome and double-metronome analyses), which are implemented by graphical user interfaces (GUIs), and list the steps needed to perform the analyses. In Section V, we conclude with a discussion of some

caveats and possible improvements to the demonstration, and how it might be adapted for use in the collection of high school and undergraduate laboratory[16–19] and classroom[20–24] investigations centered around understanding gravitational physics. Finally, in Appendix A, we describe the function of the GUI buttons used to execute the various steps of the single- and double-metronome analyses discussed previously in Section IV. [Sample pulse data files and the analysis routines are available for download from URL `https://github.com/nanograv/tabletop_pta/`.]

II. REQUIRED HARDWARE AND SOFTWARE

The metronome-microphone pulsar-timing-array demonstration requires two metronomes. Our preferred choice is Seiko model SQ50-V quartz metronomes (Figure 2), as this model has adjustable beats-per-minute (bpm) up to 208 bpm, adjustable volume, and two different tempo sounds—mode a and mode b, with mode b having a slightly higher pitch. Having two modes is helpful in distinguishing the pulses from the individual metronomes when both metronomes are on simultaneously, since the pulse shapes (profiles) are different.

FIG. 2: Two Seiko metronomes and one hand-held CAD U1 USB microphone (cable not shown) used for the demonstration.

One also needs some type of microphone, either an external USB microphone or an internal microphone, connected to a laptop that is set up to run the relevant data analysis routines (described below). We have found that the internal microphone on a MacBook Pro works best since it has ambient noise reduction, although it is somewhat inconvenient to physically move the laptop to simulate the passage of a gravitational wave. (We move the microphone in a small circle of radius $\sim$10 cm at constant speed, for reasons we will describe below.) Use of the internal laptop microphone for the stationary single-metronome analysis discussed in Section IV A may be preferred by

some. We have also used a hand-held CAD U1 USB Dynamic Recording Microphone (Figure 2), which is a little easier to maneuver for the double-metronome analysis discussed in Section IV B.

In addition, one needs an open space covering an area of about 3 m $\times$ 2 m for the placement of the two metronomes and microphone. A schematic diagram of the setup is shown in Figure 3.

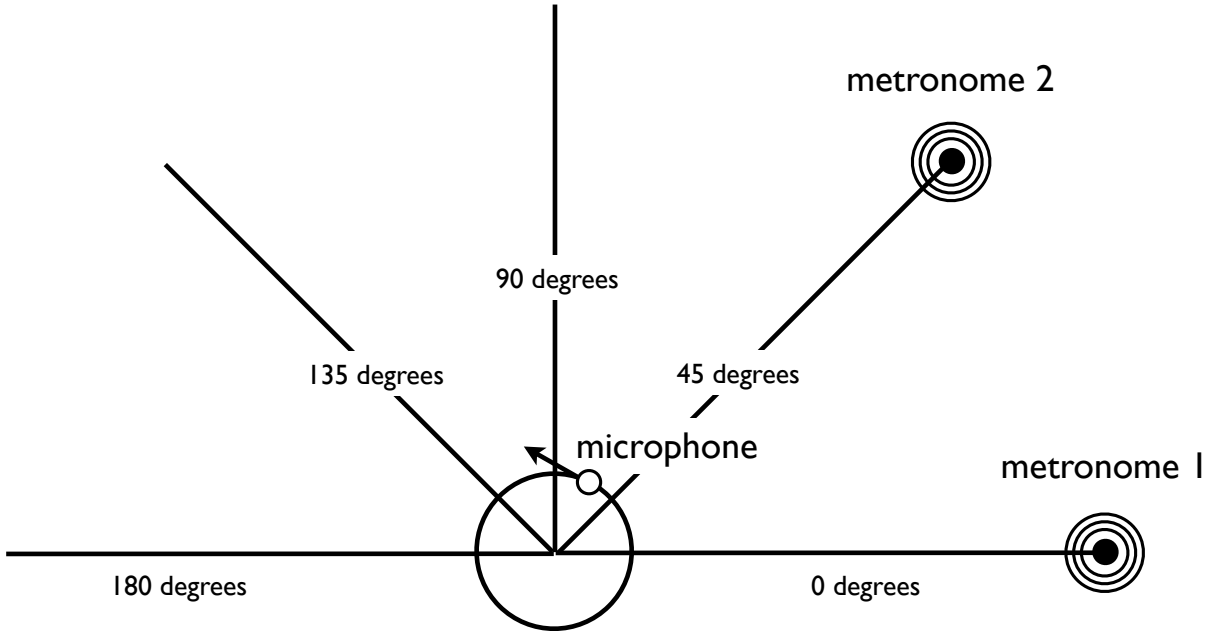

FIG. 3: Schematic diagram showing the location of the microphone and metronomes for the different analyses. The stationary microphone is located at the origin; the moving microphone undergoes uniform circular motion, indicated by the counter-clockwise circle. Metronome 2 is placed at angular location 45° in this figure, but will be placed at the other angular locations for different parts of the demonstration.

We have written Python-based routines to do the relevant data analysis calculations. These includes routines for: (i) audio recording and playback of metronome pulses, (ii) pulse data folding, (iii) pulse profile calculation, (iv) matched-filter estimation of pulse arrival times, (v) timing residual calculation, (vi) linear detrending of timing residuals, (vii) fitting of sinusoids to the timing residuals, and (viii) correlation coefficient calculation, which we describe in the next section. Two Python-based GUIs exist for performing the two main data taking and data analysis tasks: `PTAdemo_single_metronome.py` (for analyzing the single-metronome data) and `PTAdemo_double_metronome.py` (for analyzing the double-metronome data). The two GUIs and the data analysis techniques are described in more detail in the following sections.

III. PULSAR TIMING TECHNIQUES ILLUSTRATED BY THE DEMONSTRATION

The metronome-microphone demonstration is useful as an educational tool since it illustrates several techniques used in real pulsar timing analyses. It does this in the simplified context of metronome pulses recorded by a microphone, which students or the general public can more easily identify with. To set the stage for the analyses that will be described in Secs. IV A and IV B, we describe below the key techniques illustrated by the demonstration.

A. Folding data

Folding data is a technique that can be used to find both the pulse period and pulse shape (profile) in noisy time-series data[4]. The time-series (of total duration $T_{\rm tot}$) is first split into smaller segments, each of duration T, which are then averaged together. If T matches the true pulse period $T_{\rm p}$, then the pulse contributions in each segment combine coherently when the segments are summed, while the noise contributions combine incoherently (positive and negative values tending to cancel one another). If T does not match the true pulse period, the pulse contributions will effectively cancel out when averaged against the noise. So the basic procedure is to systematically try different values of T until a pulse profile "sticks out of the data", which will occur when T equals $T_{\rm p}$. The signal-to-noise ratio of the recovered pulse profile grows like $\sqrt{N}$, where N is the number of segments or, equivalently, the number of individual pulses combined[25]. Figure 4 illustrates what happens when you fold data with an incorrect and the correct pulse period.

B. Matched-filter determination of pulse arrival times

The measured times of arrival (TOAs) are determined by correlating a time-shifted version of the average pulse profile with the time-series data[26,27]. (For real pulsar timing analyses, one often uses a noiseless average "template" shape to correlate with the data. But for this demonstration we do not perform any additional denoising beyond folding the data by the pulse period to obtain the average pulse profile as discussed above.) Mathematically, one calculates the correlation function

$$C(\Delta t) \equiv \mathcal{N} \int dt\, y(t) p(t - \Delta t)\,, \tag{1}$$

where $y(t)$ is the time-series and $p(t - \Delta t)$ is the pulse profile shifted forward in time by Δt. ($\mathcal{N}$ is a normalization constant, defined below.) In the frequency domain, we have

$$C(\Delta t) = \mathcal{N} \int df\, \tilde{y}(f) \tilde{p}^*(f) e^{i2\pi f \Delta t}\,, \qquad \mathcal{N} \equiv \left[\int df\, |\tilde{p}^*(f)|^2 \right]^{-1}\,, \tag{2}$$

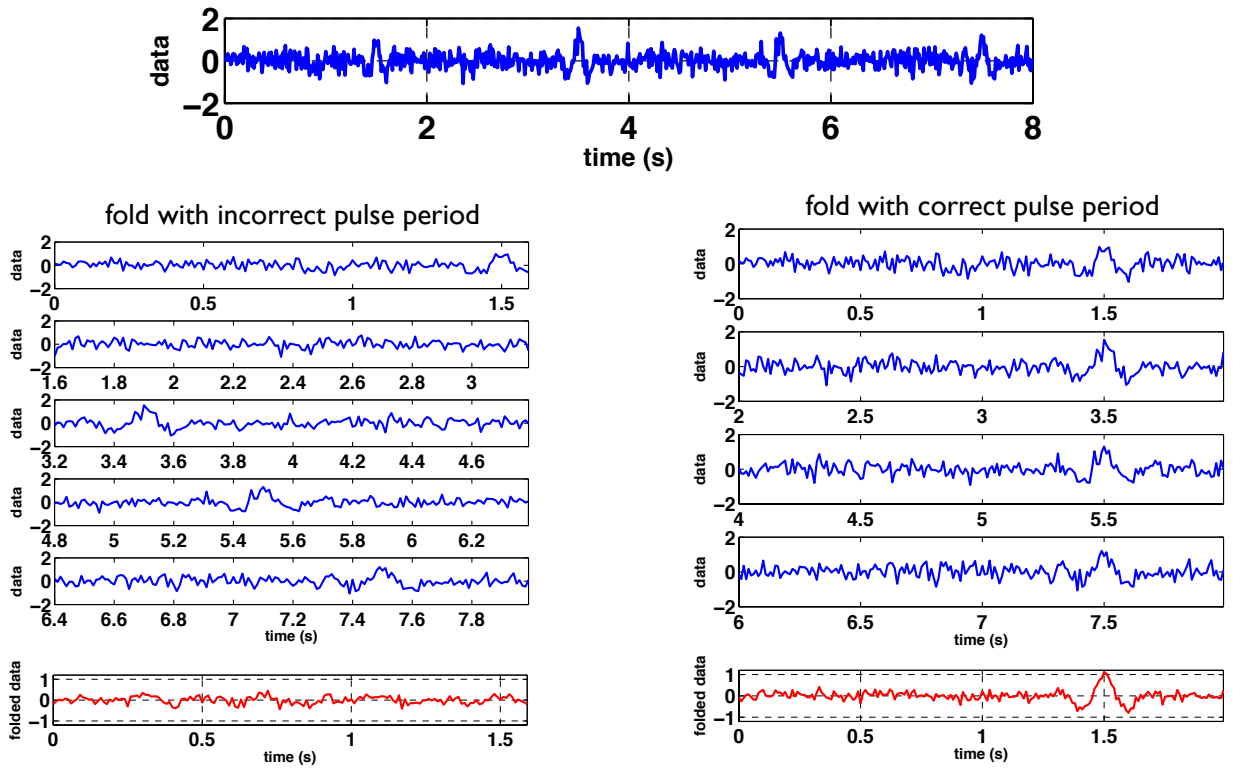

FIG. 4: Illustration of folding simulated microphone data (proportional to the acoustic wave amplitude) with both an incorrect and the correct choice for the pulse period $T_{\rm p}$. Top panel: Noisy time-series data with several injected pulses having $T_{\rm p} = 2$ s and (noiseless) pulse profile shown in the middle panel of Figure 5. Left column: Segments of the original time-series data each of duration 1.6 s (in blue), and the average of these segments (in red). Right column: Same as for the left column but for segments of duration 2 s. Note that when the data are folded with the correct pulse period $T_{\rm p} = 2$ s, the signal components combine coherently and the pulse profile is easily visible in the average of the segments (bottom right plot).

where $\tilde{y}(f)$ and $\tilde{p}(f)$ are the Fourier transforms[28] of $y(t)$ and $p(t)$. The correlation function $C(\Delta t)$ has local maxima at the arrival times of the pulses (Figure 5). In what follows, we will denote these measured arrival times by $\tau^{\rm measured}[i]$, where $i = 1, 2, \cdots$. The normalization constant $\mathcal{N}$ is included so that the values of the correlation function at the measured TOAs are estimates of the amplitudes of the pulses.

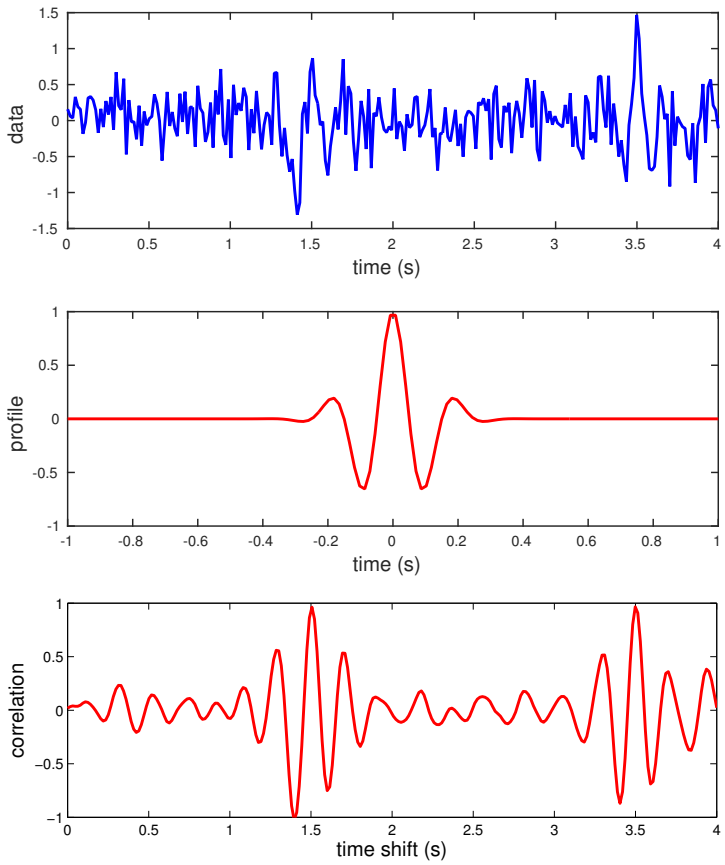

FIG. 5: Illustration on simulated microphone data showing how the correlation function has local maxima at the arrival times of the pulses. Top panel: Noisy time-series data $y(t)$ containing two pulses. Middle panel: Profile $p(t)$ of the two pulses present in the noisy data. Bottom panel: Correlation function $C(\Delta t)$ as a function of the time shift Δt (relative to $t = 0$), which has peaks at the arrival times of the two pulses, $t = 1.5$ s and $t = 3.5$ s. An animation showing the calculation of $C(\Delta t)$ as a function of the timeshift of the pulse profile is available at `https://github.com/nanograv/tabletop_pta/tree/master/docs/`, with filename `matchedfilterdemo.avi`.

C. Calculating timing residuals based on a timing model

Calculating the timing residuals is a simple matter of subtracting the expected TOAs from the measured TOAs of the pulses:

$$\delta\tau[i] = \tau^{\text{measured}}[i] - \tau^{\text{expected}}[i]\,, \tag{3}$$

where $i = 1, 2, \cdots$ labels the individual pulses. As mentioned in Sec. I, for real pulsar timing analyses the expected TOAs are determined by a rather sophisticated timing model, which takes into account the rotation period of the pulsar, its spin-down rate, its location in the sky, etc. But

for the metronome-microphone demonstration, the timing model is exceedingly simple:

$$\tau^{\rm expected}[i] = \tau^{\rm measured}[i_0] + (i - i_0)T_{\rm p}\,, \tag{4}$$

which is just the measured TOA of the pulse having the largest correlation with the pulse profile (which is indexed by i_0), plus integer multiples of the pulse period $T_{\rm p}$ of the metronome.

D. Improving the timing model by removing a linear trend in the residuals

A linear trend in the timing residuals is an indication that the pulse period determined by folding the data (Section III A) is not quite right. This is because $\delta\tau[i]$ involves a term $-(i-i_0)T_{\rm p}$, and if there is an error ϵ in $T_{\rm p}$, this term grows linearly with the pulse number i. That is, if $T'_{\rm p} = (1+\epsilon)T_{\rm p}$, then

$$\begin{aligned}
\delta\tau'[i] &= \tau^{\rm measured}[i] - \tau'^{\rm expected}[i] \\
&= \tau^{\rm measured}[i] - \left(\tau^{\rm measured}[i_0] + (i-i_0)T'_{\rm p}\right) \\
&= \tau^{\rm measured}[i] - \left(\tau^{\rm measured}[i_0] + (i-i_0)(1+\epsilon)T_{\rm p}\right) \\
&= \delta\tau[i] - \epsilon(i-i_0)T_{\rm p}\,.
\end{aligned} \tag{5}$$

By fitting a line to the timing residuals and removing this trend, we obtain an improved estimate of the pulse period, which we can then be used for subsequent timing model calculations. Figure 6 shows measured timing residuals for metronome pulses before and after removing a linear trend. (Further improved estimation of the linear drift and thus the overall stability of the metronomes might be performed by counting pulses over a longer timespan and comparing the total number of beats over, say, a several minute span.)

E. Calculating correlation coefficients between pairs of timing residuals

The correlation coefficient between a pair of timing residuals is simply the time-averaged product of the timing residuals for a pair of Earth-pulsar (or microphone-metronome) baselines. More generally, we can define the *binned correlation function* for two sets of timing-residuals $\delta\tau_1$, $\delta\tau_2$, as

$$C_{12}[k] \equiv \frac{1}{N_{\rm bins}[k]} \sum_{i,j} \delta\tau_1[i]\delta\tau_2[j]\,, \tag{6}$$

where the indices i and j for the double sum are such that $\tau_1^{\rm measured}[i] - \tau_2^{\rm measured}[j]$ is between $(k-0.5)\Delta t$ and $(k+0.5)\Delta t$, where Δt is the chosen bin size. (In other words, the condition on the

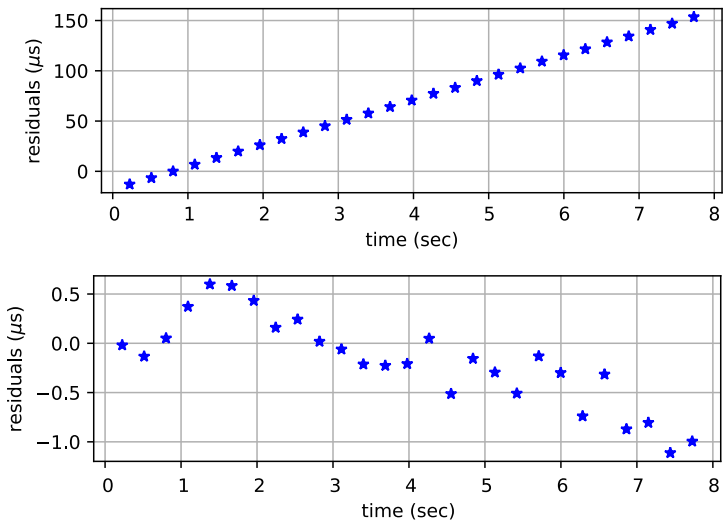

FIG. 6: Residuals for metronome pulses: before detrending (top panel) and after detrending (bottom panel). Note the different scales on the y-axis.

indices i and j is that the difference in the measured TOAs for the two timing residuals must lie in the kth lag bin.) The normalization factor $N_{\rm bins}[k]$ is simply the number of timing-residual pairs summed in the kth bin. However, for the purposes of the demonstration: (i) we are interested in only the zero-lag result (so $k = 0$), and (ii) to simplify the calculation, we can fit smooth curves $x_1(t)$, $x_2(t)$ to the two sets of timing residuals, so that

$$C_{12}[0] \quad \longrightarrow \quad \langle x_1 x_2 \rangle \equiv \frac{1}{T_{\rm tot}} \int_0^{T_{\rm tot}} dt\, x_1(t) x_2(t)\,. \tag{7}$$

Normalizing by the rms values $\sqrt{\langle x_1^2 \rangle}$ and $\sqrt{\langle x_2^2 \rangle}$, we get

$$\rho_{12} \equiv \langle x_1 x_2 \rangle / \sqrt{\langle x_1^2 \rangle \langle x_2^2 \rangle} \tag{8}$$

for the correlation coefficient, which takes values between -1 and 1. Note that for timing residuals induced by uniform circular microphone motion (see Secs. III G and III H), the best-fit smooth functions to the timing residuals will be sinusoids.

F. Microphone-motion-induced timing residuals

Similar to calculating the response of an Earth-pulsar baseline to a passing gravitational wave, one can calculate the response of a metronome timing residual to the motion of the microphone:

$$\delta\tau_I^{\rm microphone}(t) \equiv \frac{\Delta L_I(t)}{c_{\rm s}} = \frac{L_I(t) - L_I}{c_{\rm s}} \simeq -\frac{1}{c_{\rm s}} \hat{u}_I \cdot \vec{r}(t) \tag{9}$$

where $L_I(t)$ is the distance between metronome $I = 1, 2$ and the location of the microphone $\vec{r}(t)$ at time t, L_I is the nominal distance between the metronome and the microphone pointing in direction $\hat{u}_I$, and $c_{\rm s}$ is the speed of sound in air (see Figure 7). This response is just the change in

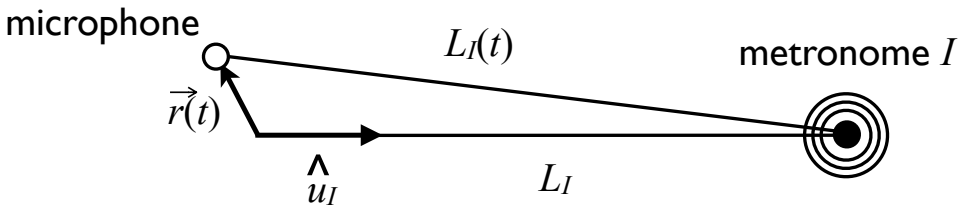

FIG. 7: Schematic diagram showing the geometrical relationship between the location of the microphone $\vec{r}(t)$ at time t and the metronome I located in direction $\hat{u}_I$ relative to the nominal position of the microphone (the origin).

the metronome pulse propagation time due to the motion of the microphone relative to metronome I. The last (approximate) equality above is valid if we ignore correction terms of order $A/L \sim 0.1$, where $A \sim 10$ cm is the amplitude of the microphone motion and $L \sim 1$ m is the distance from the metronome to the microphone when it is at the origin. Such an approximation amounts to ignoring the curvature of the pulse wavefronts.

G. Expected correlations in the timing residuals induced by uniform circular motion

For the microphone undergoing uniform circular motion with amplitude A and frequency f_0,

$$\vec{r}(t) = A\left[\cos(2\pi f_0 t + \phi_0)\,\hat{x} + \sin(2\pi f_0 t + \phi_0)\,\hat{y}\right] , \tag{10}$$

it follows immediately that

$$\delta\tau_I^{\rm microphone}(t) \simeq -\frac{A}{c_{\rm s}}\cos(2\pi f_0 t + \phi_0 - \theta_I) , \tag{11}$$

where θ_I is the angle that the position vector of metronome I makes with respect to the x-axis, and where (as before) we have ignored the higher-order correction terms to the residual. Using the trigonometric identity

$$\cos A \cos B = \frac{1}{2}\left[\cos(A+B) + \cos(A-B)\right] , \tag{12}$$

it is fairly easy to show that the time-averaged correlation coefficient of the microphone-induced timing residuals for metronomes 1 and 2 is

$$\rho_{12} \simeq \cos\zeta , \tag{13}$$

where $\zeta \equiv \theta_1 - \theta_2$ is the angle between the two microphone-metronome baselines (see Figure 8). This equality is correct to order $(A/L)^2$, where L ($\simeq L_1, L_2$) is the distance to the two metronomes.

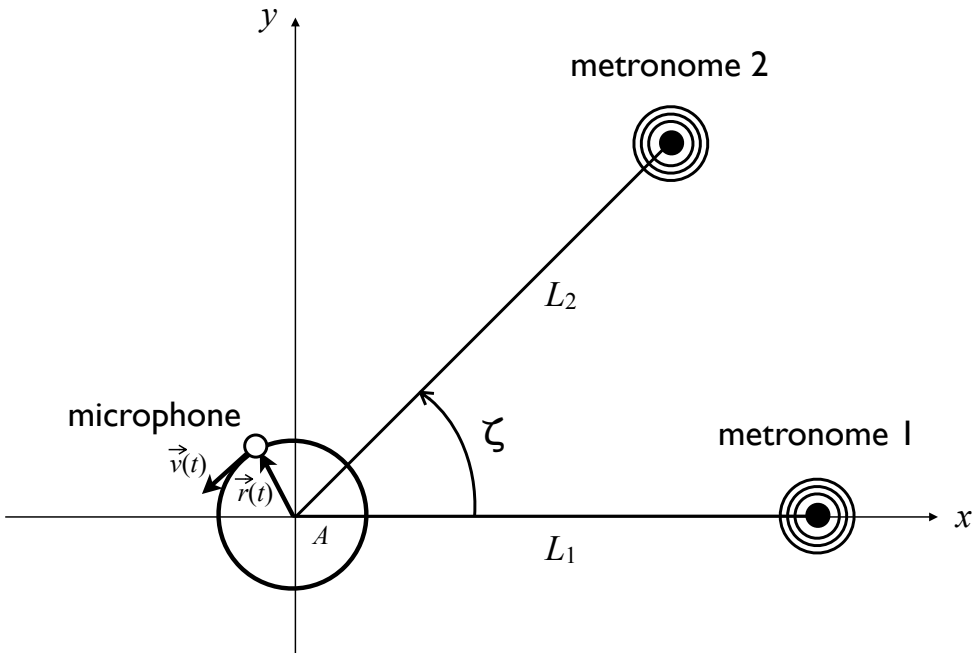

FIG. 8: Schematic diagram showing the geometrical relationship between the two metronomes and the location of the microphone $\vec{r}(t)$ at time t, which is undergoing uniform circular motion of radius A.

This dependence of the correlation coefficient on the angle between the two metronomes is what we confirm experimentally with the double-metronome analysis described in Sec. IV B. Finding the angular cosine dependence in the correlation coefficient is equivalent to finding the "smoking gun" of the Hellings and Downs curve correlation for pulsar timing arrays and thus a gravitational-wave detection. Therefore, users of this demonstration will work through the determination of the correlation themselves to illustrate their own detection of the microphone's motion. Since the timing residuals for the two metronomes typically are evaluated at different times, we actually correlate the best-fit sinusoids to the timing residuals, as described in Sec. III E.

H. Justification of the choice of uniform circular motion for the microphone

In principle, we could move the microphone in any way whatsoever, and we would still see correlations in the timing residuals associated with the two metronomes. But the form of the expected correlation will be more complicated than the simple $\rho_{12} \simeq \cos\zeta$ dependence that we found above. For example, a two-dimensional random-walk-type process would be analogous to a

stochastic background of gravitational waves. Additionally, if instead of uniform circular motion we let the microphone swing back-and-forth sinusoidally in a plane (i.e., along some line in the xy-plane) analogous to a continuous gravitational wave passing by the Earth, then the correlation coefficient will also depend on the angle that this plane makes with the x-axis. However, uniform circular motion has the nice property that the x and y components of its motion are statistically equivalent, but uncorrelated with one another (since $\cos(2\pi f_0 t + \phi_0)$ and $\sin(2\pi f_0 t + \phi_0)$ are orthogonal functions). It turns out that this is also the assumption that goes into the calculation of the Hellings and Downs correlation curve (Figure 1) for the real pulsar timing gravitational-wave case—i.e., the Hellings and Downs curve is derived under the assumption that the pulsar timing array is responding to a gravitational-wave stochastic background that is both *isotropic* (no preferred direction) and *unpolarized* (statistically equivalent and uncorrelated tensor polarization components)[2]. Since the form of the correlation found above is simple, we will focus on it in this work, though we re-emphasize that it is easy to visualize correlations regardless of the type of motion.

From a different perspective, the effect of uniform circular motion on the timing residuals is exactly what one sees in real pulsar-timing timing residuals if the yearly orbital motion of the Earth around the Sun is not taken into account in the timing models for the pulsars.

IV. DOING THE METRONOME-MICROPHONE DEMONSTRATION

In this section, we describe the GUIs used to perform the single- and double-metronome analyses for the metronome-microphone pulsar-timing-array demonstration. We show screenshots of the two GUIs in action on some sample metronome-pulse data, and describe the various steps needed to perform the analyses; a further description of the GUIs is provided in Appendix A. The ultimate goal of the demonstration is to generate a plot of the correlation coefficient ρ_{12} of the timing residuals for the metronome pulses as a function of the angular separation ζ of the two microphone-metronome baselines, and then compare that with the expected result, $\rho_{12} \simeq \cos\zeta$, for the case of uniform circular microphone motion.

A. Single-metronome analysis (stationary microphone)

The purpose of the single-metronome analysis is to calculate the pulse period and pulse profile for each metronome separately in the absence of microphone motion. The pulse periods and pulse

profiles calculated here can be considered as *reference* periods and profiles, to be used as inputs for the double-metronome analysis, where both metronomes are running simultaneously and the microphone is moving when it is recording pulses.

A screenshot of the GUI for this analysis, `PTAdemo_single_metronome.py`, is shown in Figure 9. The GUI has space for plots of: (i) pulses from the individual metronomes, (ii) pulse profiles (obtained by folding the pulse data), and (iii) timing residuals. The GUI also has several text-entry fields and buttons, whose functions are described in Appendix A 1.

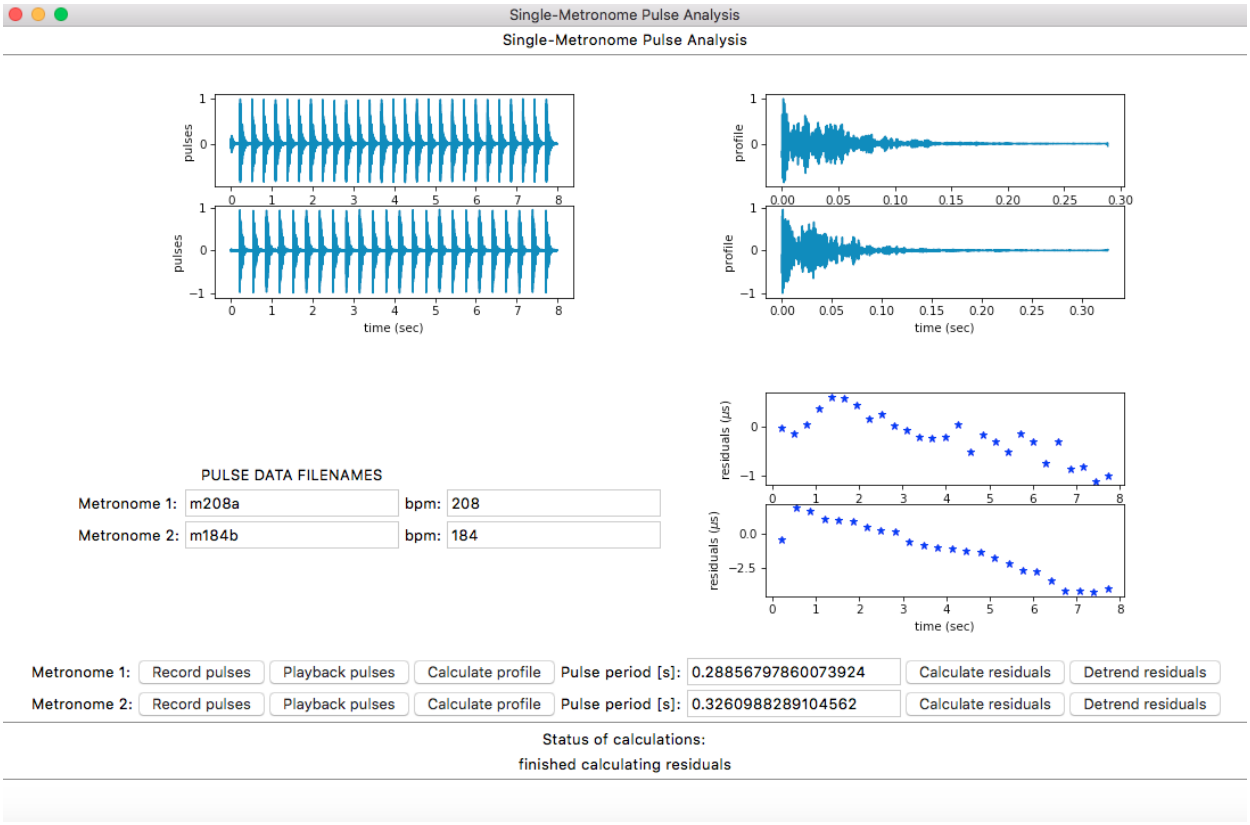

FIG. 9: Python GUI for the single-metronome analysis. Plots, text entry fields, and buttons are described in the main text and Appendix A 1.

The steps for doing the analysis are the following:

1. Record pulses from each metronome separately, keeping the microphone stationary. The microphone should be located at the origin of coordinates and the two metronomes should be at angular location 0°. The file prefixes and bpms in the text entry boxes should be chosen to match the physical settings of the metronome.

2. After recording the pulse data for each metronome, you can play it back, calculate the corresponding pulse profile and period, calculate the residuals, and detrend the residuals, by simply pressing the relevant GUI buttons.

This analysis produces two data files containing metronome pulses (e.g., `m208a.txt` and `m184b.txt`), as well as two pulse-profile data files (e.g., `m208a_profile.txt` and `m184b_profile.txt`), and the associated pulse periods for the two metronomes, which are needed for the double-metronome analysis described in the next section.

B. Double-metronome analysis (for uniform circular microphone motion)

The purpose of the double-metronome analysis is to experimentally verify the expected $\rho_{12} \simeq \cos\zeta$ correlation coefficient for the timing residuals for the two metronomes when the microphone is undergoing uniform circular motion. This is analogous to real pulsar timing analyses looking for evidence of the Hellings and Downs correlation curve when correlating timing residuals for pairs of Earth-pulsar baselines.

A screenshot of the GUI for this analysis, `PTAdemo_double_metronome.py`, is shown in Figure 10. The GUI has space for plots of: (i) pulses from the two metronomes running simultaneously, (ii) reference pulse profiles for the two metronomes, which were calculated using `PTAdemo_single_metronome.py`, and (iii) timing residuals for the two metronomes. The GUI also has several text entry fields and buttons, whose functions are described in Appendix A 2.

The steps for doing the analysis are the following:

1. Start by placing both metronomes at the same angular location 0° and at the same distance $L \sim 1$ m from the origin. With both metronomes running simultaneously, record the audio data while moving the microphone in uniform circular motion about the origin: Typically, it is best to have $A \approx 10$ cm (=0.1 m) and period of oscillation of a few seconds so that the orbit can be sampled appropriately given the periods of the metronomes. This leads to microphone-induced timing residuals of order $A/c_{\rm s} \approx 3 \times 10^{-4}$ s, where $c_{\rm s} = 340$ m/s is the speed of sound in air. The size of these residuals turns out to be more than an order-of-magnitude larger than the precision to which we can estimate the TOAs of the metronome pulses, meaning that we can easily observe the effect of microphone motion in the timing-residual data.

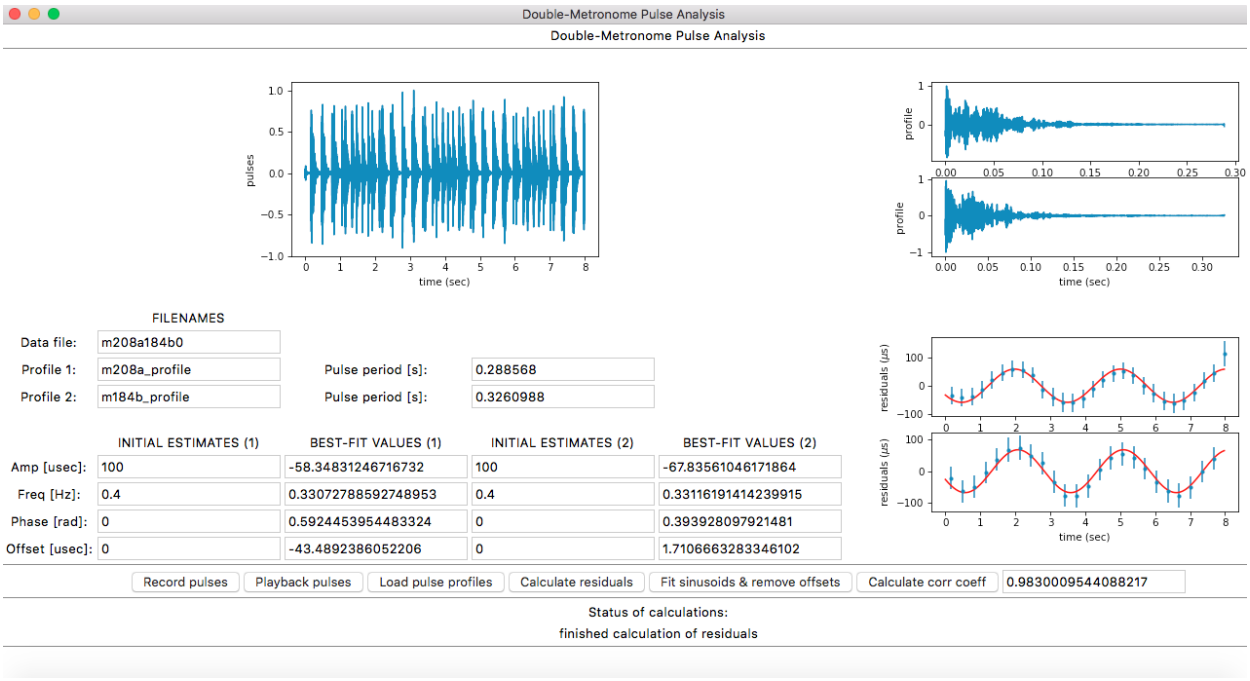

FIG. 10: Python GUI for the double-metronome analysis. Plots, text entry fields, and buttons are described in the main text and Appendix A 2.

2. After recording the double-metronome data, you can play it back, load the pulse profiles, and calculate the residuals for each metronome. The timing residuals induced by the microphone motion should be sinusoidal and have large signal-to-noise ratio. You should then fit sinusoids to the residuals for each metronome, adjusting the `INITIAL GUESS` amplitudes, frequencies, and phases if necessary. (The initial guesses just have to be close, not exact.) Finally, you should calculate the correlation coefficient, which should have a value very close to 1 for this case, since the two microphones are at the same angular location.

3. Repeat the above two steps but with metronome 2 at different angular locations (e.g., 45°, 90°, 135°, 180°) with respect to metronome 1 (which should always remain at 0°). The motion of the microphone should be as similar as possible to that for Step 1. Change the name of the file prefix in the `Data file` text entry box to `m208a184bXX`, where `XX` is `45`, `90`, `135`, `180`, to reflect the change in the angular location of metronome 2. You should find that the correlation coefficient ρ_{12} is approximately equal to $\cos\zeta$, where $\zeta = 45°, 90°, 135°, 180°$ is the angular separation of the two metronomes, in accord with Eq. 13.

This analysis produces metronome-pulse data files `m208a184bXX.txt` and values of the correlation coefficient ρ_{12} for different angular separations between the two microphone-metronome

baselines—e.g., $\zeta = 0°$, $45°$, $90°$, $135°$, $180°$ (corresponding to `XX` in the data file names). For example, the values of ρ_{12} obtained when running the correlation analysis on the data files `m208a184bXX.txt` located at `https://github.com/nanograv/tabletop_pta/tree/master/tabletop_pta/demo_data` are $\rho_{12} = 0.983, 0.776, -0.274, -0.724, -0.993$. These values are plotted in Figure 11, compared to the expected values given by $\rho_{12} \simeq \cos\zeta$ (the solid line).

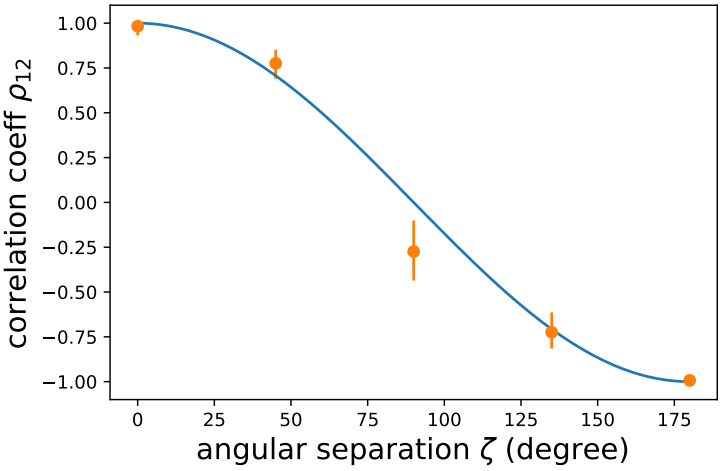

FIG. 11: Measured correlation coefficient values ρ_{12} as a function of the angular separation ζ between a pair of microphone-metronome baselines, for a particular set of demonstration pulse data files. (The error bars on the data points are determined by the method described in the text.) The solid line shows the expected dependence $\rho_{12} \simeq \cos\zeta$.

One can perform the experiment many times in order to build up statistics on the uncertainty of ρ_{12}. For this work, instead we take the best-fit sinusoid parameters (e.g., see the sine-wave fits in Figure 10 for the $\zeta = 0$ trial) with their uncertainties and then simulate many (10,000) realizations of new sinusoids for each value of ζ. We draw new parameters for the sinusoids centered around the original best-fit values and assuming Gaussian errors with standard deviations equal to each of the parameter uncertainties. For each realization, we then calculate ρ_{12} as before. In Figure 11, we show the values as listed above but include the $\pm 34.1\%$ (around the median) confidence intervals from the many trials as "realization errors" to quantify our uncertainty on our estimates of ρ_{12}.

V. DISCUSSION

We have described a demonstration using two metronomes and a microphone that serves as an acoustical analogue of a Galactic-scale gravitational-wave detector, i.e., a pulsar timing array. This

demonstration also serves as an educational tool, illustrating several techniques used in real pulsar timing analyses, but in the simplified context of metronome pulses recorded by a microphone. From our experience, we have found that the demonstration is best suited for undergraduates or senior-level high-school students who already have some familiarity with basic physics and astronomy. For less mathematically-inclined audiences, the mathematical discussion of the underlying data analysis techniques needs to be reduced accordingly. But the main idea that a common disturbance (in this case, the microphone motion) can induce correlations in the pulse arrival times, and a graphical display showing the timing residuals from the two metronomes being shifted by an amount equal to their angular separation is accessible to nearly all audiences.

A. Some caveats

The tricky technical aspect of the double-metronome analysis is to properly extract the pulse arrival times when the two metronomes are running simultaneously, producing pulses that can significantly overlap with one another (unless the pulse periods are sufficiently large and similar enough that the pulses can be spaced separately one after another). The fact that the pulse profiles $p_I(t)$ for the two metronomes ($I = 1, 2$) are different for different tempo modes a and b is crucial for distinguishing the pulses from the two metronomes. Still, the correlation functions $C_I(\Delta t)$ have several local maxima, and we need to find the largest local maxima in the vicinity of the expected pulse arrival times to determine the measured TOAs $\tau_I^{\rm measured}[i]$. If the search window is not properly centered on the expected arrival time or if it includes a local maximum of the correlation function that doesn't correspond to the true arrival time of the pulse, then the returned measured TOA will deviate from its true value, thus causing errors in the corresponding timing residual and the subsequent fit to the residuals. To help mitigate such problems, the routine that calculates the measured TOAs currently uses an adaptive width for the search window, which increases in size if it originally does not include a peak in the correlation function (this is usually a sign that the window was not large enough to include the true pulse arrival time).

Even with this adaptive-search-window technique, we sometimes do not get good agreement between the measured and theoretical correlation coefficients for intermediate separation angles between the two metronomes, i.e., ζ close to 90 degrees. A possible alternative reason for this might be reflections of the sound waves off of the table top or parts of the laptop, when using the laptop's internal microphone to do the recordings. Recovery of the expected correlation is usually better if we use a USB microphone, which does not have many intervening parts to interfere with

the sound waves.

B. Possible use as an instructional laboratory investigation

Although we have not tried to use this demonstration in its full form as an instructional laboratory, we suspect that some variant of this demonstration might be useful for an undergraduate physics or astronomy lab. We have used the single-metronome demonstration at public outreach events and for a high-school Advanced Placement Physics demonstration with good success in getting students to understand the fundamentals of pulsar timing, based on questions asked throughout. In a lab for more advanced students, the usefulness of the full demonstration comes in the form of learning the data analysis techniques of folding, matched-filtering, cross-correlation, etc., which are very general and have widespread applicability in many branches of science. Students who are comfortable with computer programming could be asked to code up their own data folding and matched-filtering routines, etc., and apply them to the metronome-pulse data. Or the students could take the routines that already exist, but create their own customized GUIs to perform custom analyses on other recorded sound files. Of course, one could simply try to use the existing demonstration (more or less as is) as a lab, but we suspect that it would be best if it were done as a "communal effort", at least as far as the metronome-pulse data taking is concerned. In other words, the two metronomes and single microphone would be shared amongst all lab groups, but each group would be responsible for performing one of the single-metronome or double-metronome analyses (e.g., a double-metronome analysis for a specific angular separation). Otherwise, there would be too much noise contamination from $\sim$10 pairs of metronomes running simultaneously.

C. Enhancements under development

To make it easier for people who are not computer savvy to perform the demonstration, we are currently developing a web-based interface for running the data analysis part of the demonstration. This will eliminate the need for the demonstrator to have a working installation of all the requisite Python routines and Python packages on his/her own computer, and should simplify the operation of the GUIs. Although for this scheme the data analysis will be done remotely on the web server, the data taking will still be done locally using the two (physical) metronomes and e.g., a smartphone for recording the pulses. The sound files recorded by the smartphone would then need to be uploaded to the web server for the subsequent single-metronome and double-metronome analyses.

Moving further in this direction, we also have an implementation of the metronome-microphone demonstration that exclusively uses readily-available smartphones to drive the demonstration[29–31]. We have written a smartphone app, called *TableTopPTA*, which allows a smartphone to operate as either a metronome or a microphone, as well as to perform all of the data analysis calculations needed for the single-metronome and double-metronome analyses. The demonstration can then be done using just three smartphones, two of which run in metronome mode; the other running in microphone mode and performing the subsequent data analysis calculations. The app is written in Javascript and runs on Android smartphones; we have not yet written an iPhone version of the app. The current code is available for public download from URL `https://github.com/marcnormandin/tabletop_pta`. Although some of the analysis routines for the smartphone app are not currently as up-to-date as those for the Python implementation of the demonstration, we have decided to make the code publicly available in case people want to experiment with what we currently have and possibly improve the code in the process.

Acknowledgments

We would like to acknowledge support from NSF Physics Frontier Center award number 1430284. JDR and MN would also like to acknowledge support from NSF grant PHY-1505861.

Appendix A: GUI buttons for the single- and double-metronome analyses

1. `PTAdemo_single_metronome.py`

Record pulses: Records audio pulse data from a metronome, and save the corresponding time-series to an ascii `.txt` file with file prefix specified by the `PULSE DATA FILENAMES` text entry boxes (default `m208a` or `m184b`). The `bpm` text entry boxes have the beats-per-minute settings for the two metronomes. The pulse recording routine is hard-coded to record 8 seconds of data.

Playback pulses: Plays back and plots the audio pulse data saved in the ascii files, again defaulted to `m208a.txt` or `m184b.txt`.

Calculate profile: Either (i) calculates the pulse period $T_{\rm p}$ and pulse profile directly by folding the metronome data, and then saves the profile to the file `m208a_profile.txt` or `m184b_profile.txt`, or (ii) reads in the pulse profile data that has already been saved

in these files. Method (i) is used only the first time the analysis is run. If the pulse profiles are read-in from the files `m208a_profile.txt` and `m184b_profile.txt`, the text entry boxes for the pulse periods need to be entered by hand. For both cases, the pulse profile is plotted from 0 to $T_{\rm p}$.

Calculate residuals: Calculates the timing residuals by subtracting the expected TOAs from the measured TOAs of the pulses as described in Secs. III C and III B.

Detrend residuals: Improves the estimate of the pulse period for a metronome by removing a linear trend from the timing residuals as described in Sec. III D. Detrending may change the estimate of the pulse period by 1-2 microseconds. The updated period is displayed in the `Pulse period` text entry box.

2. `PTAdemo_double_metronome.py`

Record pulses: Records audio pulse data from the two metronomes running simultaneously, and then saves the corresponding time-series to an ascii `.txt` file with file prefix specified by the `Data file` text entry box under the `FILENAMES` label (default `m208a184b0`).

Playback pulses: Plays back and plots the audio pulse data saved in `m208a184b0.txt`.

Load pulse profiles: Reads-in and plots the reference pulse profiles for the two metronomes, which were calculated by `PTAdemo_single_metronome.py` and were saved in ascii `.txt` files specified by the `Profile 1,2` text entry boxes under the `FILENAMES` label (default `m208a_profile` and `m184b_profile`). The text entry boxes for the pulse periods should be filled in with the values calculated by `PTAdemo_single_metronome.py`.

Calculate residuals: Calculates the timing residuals as described previously for `PTAdemo_single_metronome.py`.

Fit sinusoids & remove offsets: Simultaneously removes constant offsets and calculates best-fit sinusoids to the timing residuals for the two metronomes, using initial estimates for the amplitude, frequency, and phase of the sinusoids, and the constant offset given in the text entry boxes labeled `INITIAL ESTIMATES (1,2)` (the defaults are 100 microseconds, 0.4 Hz, 0 radians, and 0 microseconds, respectively). The constant offset arises from the arbitrariness of setting the timing residual of the pulse with the highest correlation to zero. The best-fit

parameter values calculated by `Fit sinusoids & remove offsets` are written to the text entry boxes labeled `BEST-FIT VALUES (1,2)`.

Calculate corr coeff: Calculates the time-averaged correlation coefficient between the best-fit sinusoids for the two sets of timing residuals, as described in Sec. III E. Theoretically, the value of the correlation coefficient should equally $\cos\zeta$, where ζ is the separation angle between the two microphone-metronome baselines, as described in Sec III G.

---

* Electronic address: michael.lam@mail.wvu.edu

† Electronic address: joseph.romano@utrgv.edu

‡ Electronic address: joeykey@uw.edu

§ Electronic address: marc.normandin@utrgv.edu

¶ Electronic address: jeffrey.hazboun@utrgv.edu

[1] Detweiler, S. 1979, "Pulsar Timing Measurement and the Search for Gravitational Waves", Astrophys. J. , 234, 1100

[2] Hellings, R. W., & Downs, G. S. 1983, "Upper Limits on the Isotropic Gravitational Radiation Background from Pulsar Timing Analysis", Astrophys. J. L., 265, L39

[3] Bizouard, M. A., Jenet, F., Price, R., & Will, C. M. 2013, "Pulsar Timing Arrays", Classical and Quantum Gravity, 30, 220301

[4] Lorimer, D. R., & Kramer, M. 2012, *Handbook of Pulsar Astronomy*, by D. R. Lorimer , M. Kramer, Cambridge, UK: Cambridge University Press, 2012

[5] Hobbs, G., Coles, W., Manchester, R. N., et al. 2012, "Development of a Pulsar-based Time-scale", Mon. Not. Royal Ast. Soc., 427, 2780

[6] Cordes, J. M., & Downs, G. S. 1985, "JPL Pulsar Timing Observations. III. Pulsar Rotation Fluctuations", Astrophys. J. S., 59, 343

[7] Lam, M. T., Cordes, J. M., Chatterjee, S., et al. 2017, "The NANOGrav Nine-Year Data: Excess Noise in Millisecond Pulsar Arrival Times", Astrophys. J. , 834, 35

[8] Lentati, L., Shannon, R. M., Coles, W. A., et al. 2016, "From Spin Noise to Systematics: Stochastic Processes in the First International Pulsar Timing Array Data Release", Mon. Not. Royal Ast. Soc., 458, 2161

[9] Cordes, J. M. 2013, "Limits to PTA Sensitivity: Spin Stability and Arrival Time Precision of Millisecond Pulsars", Classical and Quantum Gravity, 30, 224002

[10] Condon, J. J., & Ransom, S. M. 2016, "Essential Radio Astronomy", Princeton University Press

[11] Estabrook, F. B., & Wahlquist, H. D. 1975, "Response of Doppler Spacecraft Tracking to Gravitational Radiation", General Relativity and Gravitation, 6, 439

[12] Abbott, B. P., Abbott, R., Abbott, T. D., et al. 2016, "Observation of Gravitational Waves from a Binary Black Hole Merger", Physical Review Letters, 116, 061102

[13] https://www.ligo.caltech.edu/page/detection-companion-papers

[14] Abbott, B. P., Abbott, R., Abbott, T. D., et al. 2017, "Observation of Gravitational Waves from a Binary Neutron Star Inspiral", Physical Review Letters, 119, 161101

[15] Jenet, F. A., & Romano, J. D. 2015, "Understanding the Gravitational-Wave Hellings and Downs Curve for Pulsar Timing Arrays in Terms of Sound and Electromagnetic Waves", AJP, 83, 635

[16] Rubbo, L. J., Larson, S. L., Larson, M. B., Ingram, D. R. 2007, "Hands-on Gravitational Wave Astronomy: Extracting Astrophysical Information from Simulated Signals", AJP, 75, 597

[17] Newburgh, R. 2008, "A Demonstration of Einstein's Equivalence of Gravity and Acceleration", European Journal of Physics, 29, 2

[18] Ford, J., Stang, J., & Anderson, C. 2015, "Simulating Gravity: Dark Matter and Gravitational Lensing in the Classroom", The Physics Teacher, 53, 557

[19] Burko, L. M. 2017, "Gravitational Wave Detection in the Introductory Lab", The Physics Teacher, 55, 288

[20] Farr, B., Schelbert, G., & Trouille, L. 2012, "Gravitational Wave Science in the High School Classroom", AJP, 80, 898

[21] Kassner, K. 2015, "Classroom Reconstruction of the Schwarzschild Metric", European Journal of Physics, 36, 6

[22] Mathur, H., Brown, K., & Lowenstein, A. 2017, "An Analysis of the LIGO Discovery Based on Introductory Physics", AJP, 85, 676

[23] Kaur, T., Blair, D., Moschilla, J., Stannard, W., & Zadnik, M. 2017, "Teaching Einsteinian Physics at Schools: Part 1, Models and Analogies for Relativity", Physics Education, 52, 6

[24] Hilborn, R. C. 2018, "Gravitational Waves from Orbiting Binaries Without General Relativity", AJP, 86, 186

[25] Cordes, J. M., & Shannon, R. M. 2010, "A Measurement Model for Precision Pulsar Timing", arXiv:1010.3785

[26] Taylor, J. H. 1992, "Pulsar Timing and Relativistic Gravity", Royal Soc. of London Phil. Trans. Series A, 341, 117

[27] Lam, M. T., Cordes, J. M., Chatterjee, S., et al. 2016, "The NANOGrav Nine-Year Data Set: Noise Budget for Pulsar Arrival Times on Intraday Timescales", Astrophys. J. , 819, 155

[28] The Fourier transform $\tilde{y}(f)$ of the time-series $y(t)$ is defined by $\tilde{y}(f) = \int dt\, y(t) e^{-i2\pi ft}$ or, equivalently, $y(t) = \int df\, \tilde{y}(f) e^{i2\pi ft}$.

[29] Kuhn, J., Vogt, P. 2013, "Applications and Examples of Experiments with Mobile Phones and Smartphones in Physics Lessons", Frontiers in Sensors, 1, 4

[30] Martinez, L. & Garaizar, P. 2014, "Learning Physics Down a Slide: A Set of Experiments to Measure Reality Through Smartphone Sensors", International Journal of Interactive Mobile Technologies, 8, 3

[31] Osorio, M., Pereyra, C. J., Gau, D. L., & Laguarda, A. 2018, "Measuring and characterizing beat phenomena with a smartphone", European Journal of Physics, 39, 2