

Primer Informe de laboratorio

Experimento de Phyphox

Luis Carlos Leal Gamboa
est.luis.cleal@unimilitar.edu.co
 Docente: José De Jesús Rugeles

Resumen— En este informe se abordarán un par de experimentos haciendo uso de la aplicación de Phyphox para la toma de datos, estableciendo los experimentos a realizar a partir de las especificaciones presentes en el dispositivo a utilizar, además se abordarán distintos programas como lo es Matlab y Thonny los cuales nos permitirán observar los resultados obtenidos de forma de graficas a partir de los datos recopilados por la aplicación de Phyphox, de manera que sea comparable con respecto a la realidad.

Abstract— This report will address a couple of experiments using the Phyphox application for data collection, establishing the experiments to be carried out based on the specifications present in the device to be used. In addition, different programs such as Matlab and Thonny, which will allow us to observe the results obtained in the form of graphs from the data collected by the Phyphox application, so that they can be compared with reality.

DESARROLLO EXPERIMENTO

Con el fin de verificar el correcto desarrollo del experimento, se verificarán las cualidades que presenta el dispositivo a usar, en este caso un iPhone 13, para ello se visitara la página oficial de la aplicación Phyphox (<https://phyphox.org/sensordb/>), el dispositivo cuenta con los siguientes sensores:

- Sensor de acelerómetro
- Giroscopio
- Presión
- Aceleración (sin g)
- Magnetómetro
- Proximidad

Además, con lo que no cuenta el celular es con los sensores de humedad, temperatura y luz, para corroborar esto, se obtiene la siguiente figura, que corresponde directamente a la base de datos de la aplicación.

Manufacturer	Model	Accelerometer	Acceleratio...	Gyroscope	Magnetometer	Pr...	Te...	H...	Li...	Pr...
filter column...	iphone 13	Variables								
		Sample size	Rate	Rate	Rate	Rate	Rate	Rate	Rate	Rate
		Available	Available	Available	Available	Available	Available	Available	Available	Available
Apple	iPhone13 (iPhone14,5)	410	1 ✓	100.0 Hz	0.021 ms ✓	100.0 Hz	100.0 Hz	100.0 Hz	100.0 Hz	100.0 Hz

Fig. 1. Características del dispositivo utilizado.

A partir de lo evidenciado en la figura número 1 se determinan los experimentos posibles para realizar, en este caso se realizan.

1. Aceleración (sin g).
2. Colisión inelástica.

Principalmente se realiza el experimento número 1, el cual corresponde a la aceleración (sin g), para ello se realiza la toma de datos correspondiente en la aplicación, en este caso se selecciona la casilla con el mismo nombre

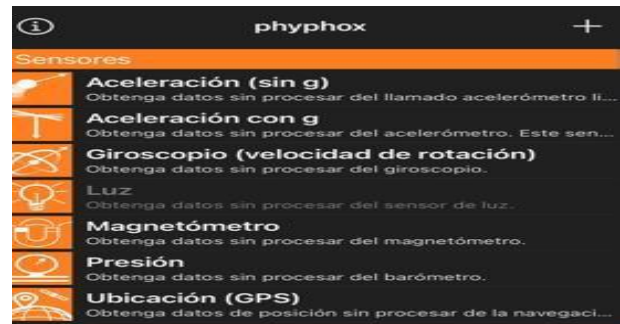


Fig. 2. Interfaz de la aplicación.

Como se observa en la figura 2, se selecciona la primera opción del interfaz de la aplicación, en este caso el experimento a realizar calcula la aceleración en cada uno de los ejes al movimiento del celular, por lo cual se observarán tres graficas las cuales determinan la aceleración en cada uno de los ejes con respecto al tiempo, como se observa a continuación.



Fig. 3. Interfaz del experimento aceleración sin fuerza g.

Como se observa en la figura número 3 la interfaz para tanto la toma de datos, como resultados del experimento, se basa en cuatro componentes importantes, donde podemos observar las tres graficas que garantizan el comportamiento de la aceleración en cada uno de los ejes con respecto al tiempo y así mismo el botón de comienzo, el cual tienen una flecha horizontal, simbolizando el “play” para de esta manera el celular comenzar a capturar los datos.

A partir de esto se obtienen los siguientes resultados.

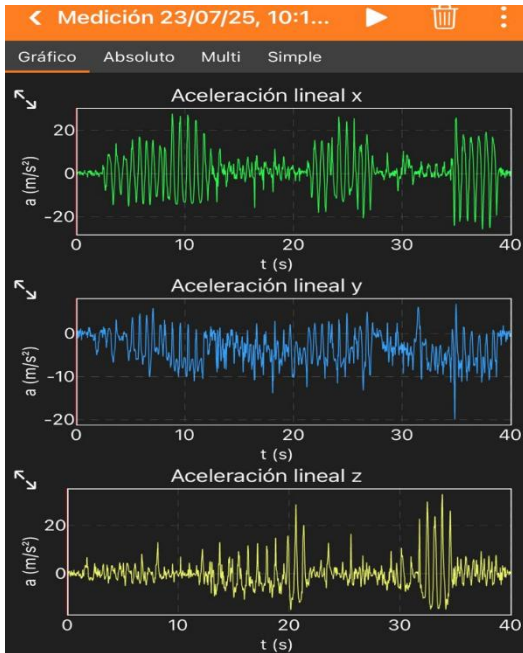


Fig. 4. Resultados obtenidos a partir del experimento aceleración sin fuerza g.

Como se observa en la figura 4, los datos obtenidos a partir de las mediciones del celular se obtienen tres graficas, las cuales representan la aceleración en cada uno de los ejes (x, y, z) con respecto del tiempo.

Así mismo se pueden extraer los datos de la aplicación con el fin de analizarlos haciendo uso de herramientas como Matlab y Thonny.

1. Matlab.

A partir de los datos obtenidos en el experimento, se añaden al sitio de Matlab con el fin de cargar los archivos y los datos presentes, se analizan, permitiendo generar así, la tabla con cada uno de los datos obtenidos, en este caso valores para la aceleración en x, y, z además de la aceleración total obtenida en el celular.

A partir de estos datos, se establece cada una de las columnas presentes en la tabla como una variable, dándole el valor de “x” como se observa a continuación.

```
x = RawData.LinearAccelerationX_m_s_2_;
```

De esta manera se realiza para todos los datos obtenidos por el celular para este eje, así consecutivamente con cada una de las aceleraciones y el tiempo.

Con el fin de graficar cada una de las aceleraciones, se determina el siguiente código, con el cual se obtendrán las gráficas de la manera más similar a la aplicación de Phyphox

```
subplot(4,1,1)
plot(t, x, 'Color', 'g')
grid on
xlabel('t(s)')
ylabel('a(m/s^2)')
title('Aceleración en el eje x')
```

A partir de estos comandos se repite para cada una de las gráficas, obteniendo el siguiente plano.

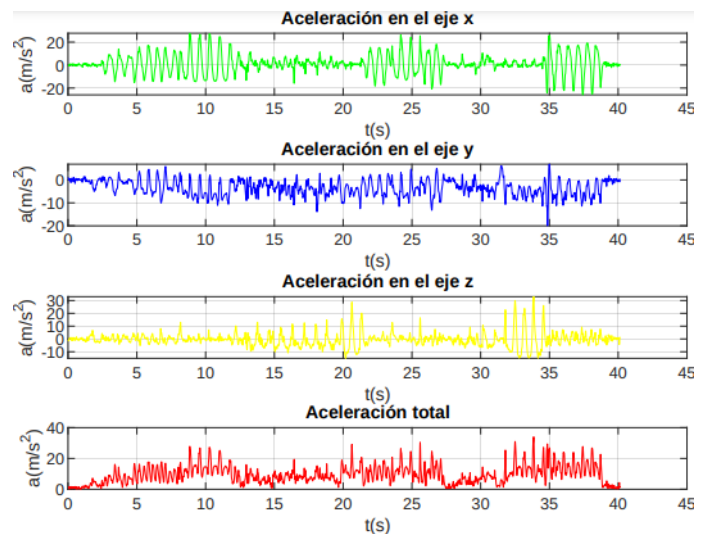


Fig. 5. Graficas obtenidas en Matlab.

Como se observa en la figura 5 se obtuvieron las mismas graficas observadas en la aplicación, además obteniendo la gráfica correspondiente para la aceleración total.

2. Thonny.

Así mismo como se realizó el procedimiento en la aplicación de Matlab, se realizó a partir de Código Python que nos dará como resultado las mismas graficas correspondientes a las obtenidas en el software de Matlab.

Para ello, es necesario determinar las siguientes librerías en la aplicación de Thonny, esto siendo fundamental para poder programar de manera que sea lo más similar al software de Matlab.

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

A partir de estas librerías, se realiza un procedimiento muy similar al realizado en Matlab, a diferencia, que, en este ámbito, es necesario llamar el archivo que contiene los datos desde la ruta específica donde este se almacene.

A partir de esta configuración realizada se obtienen las siguientes graficas.

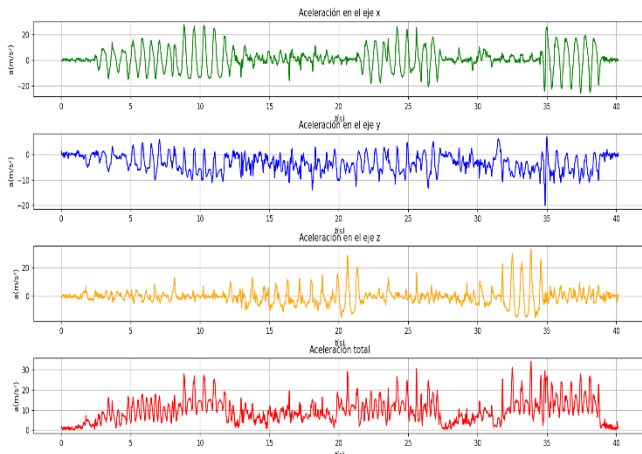


Fig. 6. Gráficas obtenidas en Thonny.

Como se observa en la figura 6 las señales obtenidas tienen mejor imagen y más calidad con diferencia a las gráficas obtenidas en el software de Matlab, pero así mismo ambas gráficas, poseen las mismas gráficas, manteniendo la similitud con respecto a los datos obtenidos en la aplicación de Phyphox. Dando a entender que el proceso realizado fue asertivo y lo más adecuado posible, evidenciando la eficacia de ambos programas para generar gráficas.

2. Amplitud de audio.

El procedimiento realizado consiste en calcular el nivel de dB que produce una copa de vidrio, para esto se hará uso del sensor de amplitud de audio presente en la aplicación de Phyphox, para ello es necesario calibrar el sensor como se ve a continuación.

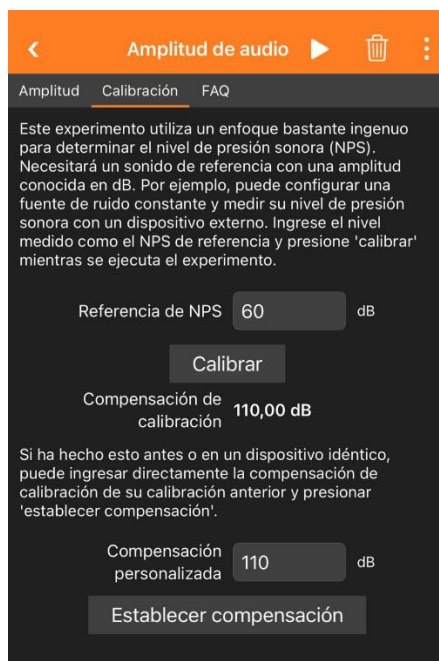


Fig. 7. Calibración del sensor de amplitud del teléfono.

Como se observa en la figura 7, la calibración se realizó haciendo uso de una estimación de compensación de audio con

respecto al ambiente, esto debido a que el sensor media un potencia en el ambiente de -70 dB aproximadamente, cuando la potencia en el ambiente de una habitación en silencio ronda una potencia de 40 dB, por lo cual la compensación de audio establecida fue de 110 dB, a partir de esto, se comienza a medir la potencia establecida por la copa de vidrio al rozar con un elemento húmedo en este caso un dedo, el borde de la copa generando así la potencia ejercida por esta, por lo cual los resultados obtenidos son.



Fig. 8. Potencia de audio ejercida por la copa.

Como se observa en la figura 8, la potencia en dB que corresponde a la copa de vidrio es de aproximadamente 90 dB esto se comprueba al momento de parar de rozar el borde de esta, así mismo reanudando el roce, se produce nuevamente una frecuencia muy similar a la anterior.

A partir de los datos obtenidos por la aplicación de Phyphox, se realiza la grafica obtenida en el software de Matlab, así mismo generando un código de Python en la aplicación de Thonny, como se observa a continuación.

Para el caso de la simulación en Matlab, se realiza un procedimiento similar al ya hecho en la simulación anterior, por lo cual se determina la variable del tiempo, como la variable de amplitud de la señal como se observa a continuación.

```
t=Amplitudes.Time_s_
dB=Amplitudes.SoundPressureLevel_dB_
```

Así mismo se realiza la creación de la gráfica.

```
plot(t,dB,Color="r")
grid on
xlabel('t(s)')
ylabel('NPS(dB)')
```

Por ende, la gráfica obtenida es.

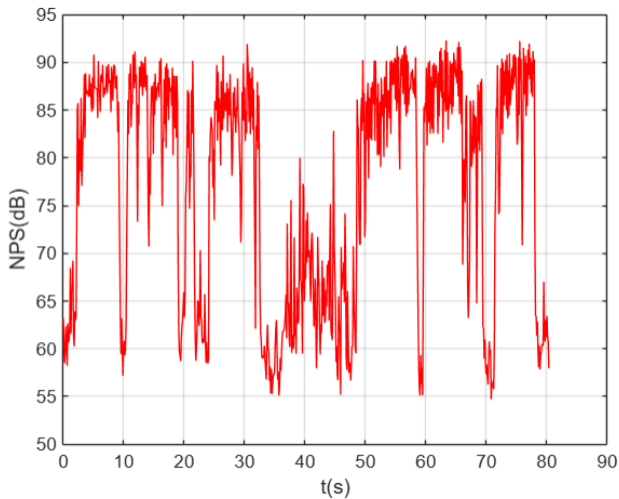


Fig. 9. Señal de la potencia ejercida por la copa realizada en Matlab.

Como se observa en la imagen 9, la señal obtenida a partir de los datos de la aplicación de Phyphox, corresponde directamente a lo obtenido por esta misma aplicación, lo que nuevamente nos reafirma la potencia en dB, que llega a ejercer el sonido de la copa de vidrio, por otro lado, se busca además garantizar el mismo resultado, pero esta vez en el programa de Thonny se realiza un código muy similar al utilizado en Matlab, a diferencia de que es necesario que el programa lea el archivo desde la ubicación de este, como se ve a continuación.

```
documento_csv = r'C:/Users/Luizl/Downloads/Amplitudes.csv'
```

Así mismo se determinan las columnas presentes, para de esta manera graficar la señal obtenida a partir de los datos.

```
dB = a['Sound pressure level (dB)']
t = a['Time (s)']
```

Así mismo se genera el comando “plot” el cual graficará con las especificaciones dadas la señal deseada.

```
plt.plot(t, dB, color='Red')
plt.grid(True)
plt.xlabel('t(s)')
plt.ylabel('NPS(dB)')
```

Por ende, a partir de esta configuración, se obtiene la siguiente señal simulada en el programa Thonny.

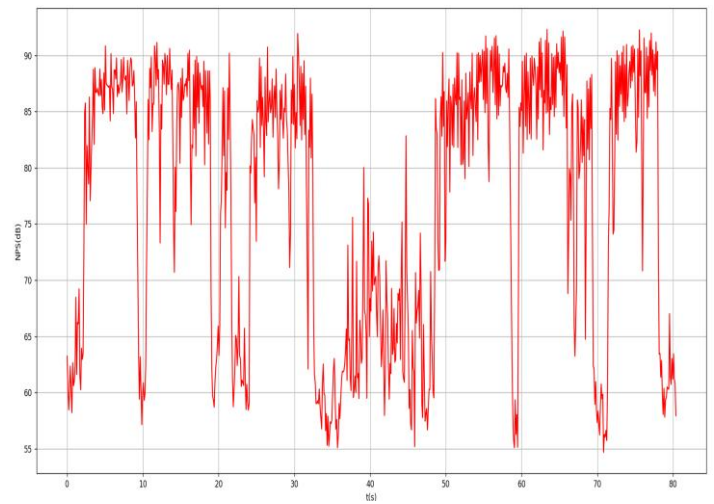


Fig. 10. Señal de la potencia ejercida por la copa realizada en Thonny.

Como se observa en la figura 10, las graficas obtenidas corresponden directamente a lo planteado por la aplicación de Phyphox, así mismo manejando todas las características de la grafica obtenida en el software de Matlab, por lo cual podemos indicar que el sonido generado por la copa tiene directamente una potencia de aproximadamente 90 dB, que es directamente proporcional al sonido ejercido, ya que este sonido es bastante fuerte e incómodo, afectando evidentemente a los oídos humanos.

Conclusiones.

- Los experimentos realizados demuestran que phyphox es efectiva para la toma de datos utilizando los sensores del iphone 13. en el experimento de aceleración (sin g), se verificó el correcto funcionamiento de los sensores en los tres ejes, resultados que fueron corroborados mediante análisis en matlab y thonny, siendo este último el que presentó mejor calidad gráfica.
- Los resultados obtenidos muestran que el sonido producido por la copa de vidrio al rozar su borde con un dedo húmedo genera aproximadamente 90 dB, valor que se mantuvo constante al repetir el roce en el borde de la copa, confirmando la estabilidad y repetibilidad del sistema de medición.
- Al analizar los datos en Matlab y Thonny, se observó que ambos programas procesaron correctamente los datos obtenidos por Phyphox, reproduciendo gráficas idénticas a las mostradas en la aplicación. Las gráficas obtenidas en Thonny presentaron mejor calidad visual que las de Matlab, aunque ambas mantuvieron la fidelidad con respecto a los datos originales.

REFERENCIAS

- [1] Staacks, S. (s. f.). *Your smartphone is a mobile lab*. Phyphox.
<https://phyphox.org/>
- [2] phyphox. (2018, 24 marzo). *Smartphone-Experiment: (In)elastic Collision (en)* [Video]. YouTube.
<https://www.youtube.com/watch?v=ikvtPDwV1FE>
- [3] *MATLAB - El lenguaje del cálculo técnico*. (s. f.).
<https://la.mathworks.com/products/matlab.html>
- [4] Thonny, Python IDE for beginners. (s. f.). <https://thonny.org/>