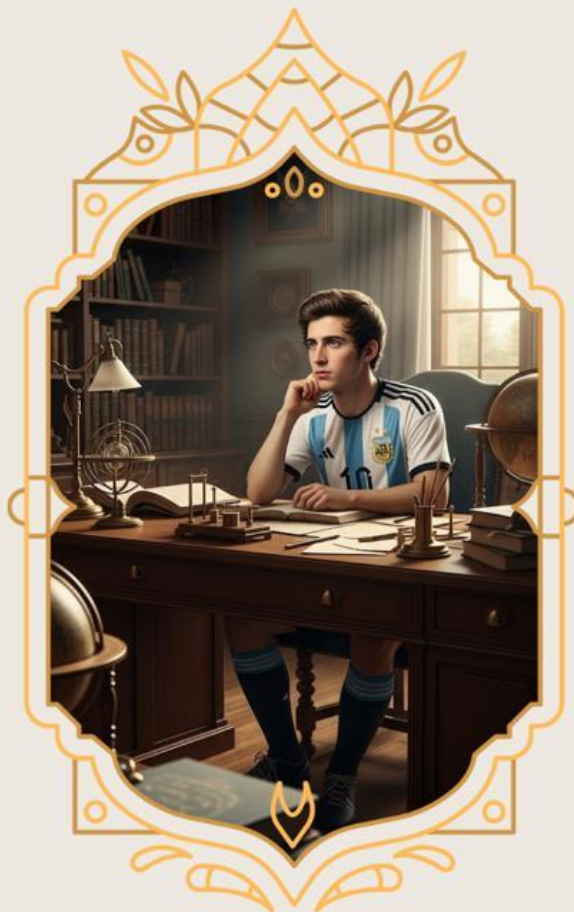


PROYECTO FINAL DE INFORMATICA

Efecto Doppler



Rosario Lujan Urtiaga Colombatti

Julio 1025

LA FÍSICA COMPUTACIONAL Y EL EFECTO DOPPLER

RESUMEN:

En el informe desarrollaremos teóricamente este fenómeno para poder comprenderlo y así generar un programa que nos simplifique la resolución de situaciones problemáticas. Para desarrollar el programa haremos uso de los conocimientos aprendidos en la asignatura, Física Computacional de la Licenciatura en Física de la UNLPam, del año 2025.

PALABRAS CLAVES: Efecto Doppler – Python

INTRODUCCION:

El Efecto Doppler es un efecto omnipresente en nuestra vida diaria, manifestándose, por ejemplo, en el cambio de tono de la sirena de una ambulancia al acercarse y alejarse, detectar el movimiento de la sangre en el cuerpo humano a través de la ecografía Doppler, entre muchos fenómenos más.

Es un fenómeno físico fascinante que describe el cambio aparente en la frecuencia de una onda (sonido o luz) percibida por un observador cuando la fuente de la onda y el observador están en movimiento relativo entre sí.

El presente trabajo aborda el estudio del Efecto Doppler, centrándose en los escenarios clave donde la **fente de la onda es fija y el observador se mueve**, donde el **observador es fijo y la fuente se mueve**, o bien, donde **ambos se encuentran en movimiento**. Para facilitar la comprensión y el cálculo de la frecuencia percibida en estas diversas situaciones, se ha desarrollado un programa en Python.

DESARROLLO:

A continuación, vamos a analizar los casos posibles que pueden existir entre la relación observador y fuente, limitándonos a ondas sonoras:

- 1) Observador móvil, fuente en reposo

Adoptemos un marco de referencia en reposo en el medio a través del cual viaja el sonido. Contaremos con una fuente de sonido S en reposo y un observador O que se mueve hacia la fuente con una velocidad v_0 .

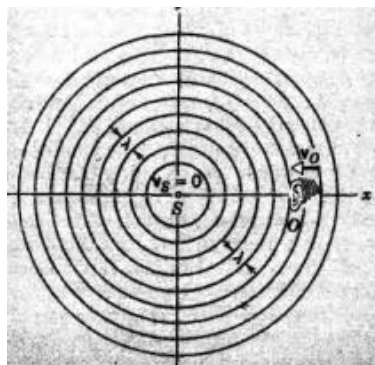


Ilustración 1 Observador en movimiento y fuente en reposo

Los círculos representan frentes de onda, separados por la distancia de una longitud de onda, que viaja a través del medio. Un observador en reposo en el medio recibiría $\frac{vt}{\lambda}$ ondas



en el tiempo t , donde v es la velocidad del sonido en el medio y λ es la longitud de onda. Sin embargo, a causa del movimiento hacia la fuente, el observador recibe $\frac{v_0 t}{\lambda}$ ondas adicionales en este mismo tiempo t . La frecuencia γ' (la elección de la letra es para que se diferencie de las velocidades) que se oye realmente es el número de ondas recibidas por unidad de tiempo, o sea

$$\gamma' = \frac{\frac{vt}{\lambda} + \frac{v_0 t}{\lambda}}{t} = \frac{v + v_0}{\lambda} = \frac{v + v_0}{\frac{v}{\gamma}}$$

Esto es,

$$\gamma' = \gamma \frac{v + v_0}{v} = \gamma \left(1 + \frac{v_0}{v}\right)$$

La frecuencia γ' captada por el oído del observador es la frecuencia γ oída en reposo más el incremento $\gamma(v_0/v)$ que surge del movimiento del observador. Cuando el observador está móvil **alejándose** de la fuente estacionaria, existe una disminución de la frecuencia $\gamma(v_0/v)$ correspondiente a las ondas que no llegan al observador en cada unidad de tiempo a causa del alejamiento. Entonces

$$\gamma' = \gamma \frac{v - v_0}{v} = \gamma \left(1 - \frac{v_0}{v}\right)$$

De aquí que la relación general que prevalece cuando la fuente está en reposo respecto al medio, pero el observador se mueve a través de él sea:

$$\gamma' = \gamma \frac{v \pm v_0}{v}$$

donde el signo más se tiene para el movimiento hacia la fuente y el signo menos se tiene para el movimiento que se aleja de la fuente. Nótese que el cambio de frecuencia ocurre porque el observador intercepta más o menos ondas en cada segundo como resultado del movimiento a través del medio.

- 2) Fuente móvil, observador en reposo

Cuando la fuente está móvil **hacia un observador** estacionario, el efecto es un acortamiento de la longitud de onda:

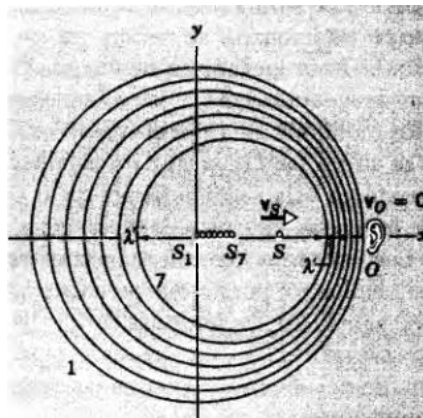


Ilustración 2: Fuente móvil y observador en reposo



Ya que la fuente está viajando tras las ondas que se aproximan, y, por lo tanto, las crestas se juntan más entre sí. Si la frecuencia de la fuente es γ y su velocidad es v_s , entonces durante cada vibración viaja una distancia (v_s / γ) , y cada longitud de onda se acorta en esta cantidad. De aquí que la longitud de onda del sonido que llega al observador no sea $\lambda = v / \gamma$, sino $\lambda = \frac{v}{\gamma} - \frac{v_s}{\gamma}$. La frecuencia del sonido que el observador oye aumenta y está dada por:

$$\gamma' = \frac{v}{\lambda'} = \frac{v}{\frac{v - v_s}{\gamma}} = \gamma \frac{v}{v - v_s}$$

Si la fuente se **mueve alejándose** del observador, la longitud de onda emitida es v_s / γ mayor que λ , de modo que el observador oye una frecuencia disminuida, es decir

$$\gamma' = \frac{v}{\frac{v + v_s}{\gamma}} = \gamma \frac{v}{v + v_s}$$

De aquí que la relación general que prevalece cuando el observador está en reposo respecto al medio, pero la fuente se mueve a través de él sea

$$\gamma' = \gamma \frac{v}{v \pm v_s}$$

donde el signo menos rige para el movimiento hacia el observador y el signo más para el movimiento alejándose del observador. Nótese que el cambio aquí es el acortamiento o el aumento de la longitud de onda transmitida en el medio debido al movimiento de la fuente en el medio.

- 3) Fuente y observador en movimiento:

Si tanto la fuente como el observador se mueven en el medio transmisor, puede demostrarse que el observador oye una frecuencia dada por

$$\gamma' = \gamma \frac{v \pm v_o}{v \mp v_s}$$

donde los signos superiores (+ en el numerador, - en el denominador) corresponden a la fuente y al observador cuando se acercan, y los signos inferiores cuando se aleja uno del otro.

Una vez entendido el funcionamiento del Efecto Doppler y las relaciones existentes entre el observador y la fuente de emisión de sonido vamos a introducirnos en Python para poder resolver problemas relacionados a este fenómeno. El objetivo del programa es poder brindarle al usuario la libertad de personalizar las características del observador y la fuente, en problemas de Efecto Doppler.

Veamos cómo funciona:

Lo primero que hace el programa es pedir que ingresemos la frecuencia de la fuente y la velocidad del sonido en el medio.

```
Calculadora del Efecto Doppler: calculamos la frecuencia percibida por un observador
Ingrese la frecuencia de la fuente (Hz): 400
Ingrese la velocidad del sonido en el medio (m/s): |
```



- Para tener de referencia, una persona percibe sonidos de entre 20 y 20.000 Hz. Un sonido con una frecuencia de 50 Hz se considera grave, mientras que uno de 20,000 Hz se considera agudo.
- Referencias de velocidad de ondas sonoras en distintos medios materiales:
 - **Aire (a 20°C):** 343 m/s
 - **Agua (a 25°C):** 1593 m/s
 - **Madera:** 3700 m/s
 - **Hormigón:** 4000 m/s
 - **Acero:** 6100 m/s

```
Escenarios posibles
1) Fuente fija, observador en movimiento
2) Observador fijo, fuente en movimiento
3) Ambos en movimiento
Seleccione el número del escenario: |
```

En esta parte de la ejecución del código podemos elegir el escenario que deseamos simular para calcular la frecuencia percibida por el observador.

```
Seleccione el número del escenario: 3
Ingrese la velocidad del observador (m/s): 20
¿El observador se ACERCA o se ALEJA de la fuente? (acerca/aleja): acerca
Ingrese la velocidad de la fuente (m/s): 10
¿La fuente se ACERCA o se ALEJA del observador? (acerca/aleja): aleja
```

En este caso elegí el escenario 3, que es el más complejo ya que tanto el observador como la fuente están en movimiento, entonces debemos ingresar y ajustar las direcciones para el observador y fuente. En los escenarios 1) y 2) como se mueve solo el observador o fuente, ingresamos y ajustamos las direcciones para una de nuestras variables.

Una vez ingresados los datos, el programa nos devolverá la frecuencia percibida por el observador.

```
La frecuencia percibida por el observador es: 411.3314447592068 Hz
¡Gracias por usar la calculadora del Efecto Doppler!
```



ANEXO

1) CODIGO

```
def efecto_doppler():  
    print("Calculadora del Efecto Doppler: calculamos la frecuencia percibida por un observador")  
  
    # Solicitar datos al usuario  
    try:  
        frecuencia_fuente = float(input("Ingrese la frecuencia de la fuente (Hz): "))  
        velocidad_sonido = float(input("Ingrese la velocidad del sonido en el medio (m/s): "))  
    except ValueError:  
        print("Error: Por favor, ingrese valores numéricos válidos.")  
  
    print("Escenarios posibles")  
    print("1) Fuente fija, observador en movimiento")  
    print("2) Observador fijo, fuente en movimiento")  
    print("3) Ambos en movimiento")  
    escenario = input("Seleccione el número del escenario: ")  
  
    if escenario == '1':  
        # Fuente fija, observador en movimiento  
        try:  
            velocidad_observador = float(input("Ingrese la velocidad del observador (m/s): "))  
            direccion_observador = input("¿El observador se ACERCA o se ALEJA de la fuente? (acerca/aleja): ")  
        except ValueError:  
            print("Error: Por favor, ingrese un valor numérico válido para la velocidad.")  
            return  
        if direccion_observador not in ['acerca', 'aleja']:  
            print("Error: La dirección del observador debe ser 'acerca' o 'aleja'.")  
            return  
  
        # Ajuste de signo para la velocidad del observador  
        if direccion_observador == 'acerca':  
            velocidad_observador_con_signo = velocidad_observador  
        else: # se aleja  
            velocidad_observador_con_signo = -velocidad_observador  
  
        # Fórmula Doppler para observador en movimiento (fuente fija)  
        frecuencia_observador = frecuencia_fuente * ((velocidad_sonido + velocidad_observador_con_signo) / velocidad_sonido)  
  
    elif escenario == '2':  
        # Observador fijo, fuente en movimiento  
        try:  
            velocidad_fuente = float(input("Ingrese la velocidad de la fuente (m/s): "))  
            direccion_fuente = input("¿La fuente se ACERCA o se ALEJA del observador? (acerca/aleja): ")  
        except ValueError:  
            print("Error: Por favor, ingrese un valor numérico válido para la velocidad.")  
            return  
        if direccion_fuente not in ['acerca', 'aleja']:  
            print("Error: La dirección de la fuente debe ser 'acerca' o 'aleja'.")  
            return  
  
        # Ajuste de signo para la velocidad de la fuente  
        if direccion_fuente == 'acerca':  
            velocidad_fuente_con_signo = -velocidad_fuente  
        else: # se aleja  
            velocidad_fuente_con_signo = velocidad_fuente  
  
        # Fórmula Doppler para fuente en movimiento (observador fijo)  
        frecuencia_observador = frecuencia_fuente * (velocidad_sonido / (velocidad_sonido + velocidad_fuente_con_signo))
```



```
elif escenario == '3':  
    # Ambos en movimiento  
    try:  
        velocidad_observador = float(input("Ingrese la velocidad del observador (m/s): "))  
        direccion_observador = input("¿El observador se ACERCA o se ALEJA de la fuente? (acerca/aleja): ")  
        velocidad_fuente = float(input("Ingrese la velocidad de la fuente (m/s): "))  
        direccion_fuente = input("¿La fuente se ACERCA o se ALEJA del observador? (acerca/aleja): ")  
    except ValueError:  
        print("Error: Por favor, ingrese valores numéricos válidos para las velocidades.")  
        return  
    if direccion_observador not in ['acerca', 'aleja'] or direccion_fuente not in ['acerca', 'aleja']:  
        print("Error: Las direcciones deben ser 'acerca' o 'aleja'.")  
        return  
    # Ajuste de signos para ambas velocidades  
    if direccion_observador == 'acerca':  
        velocidad_observador_con_signo = velocidad_observador  
    else:  
        velocidad_observador_con_signo = -velocidad_observador  
    if direccion_fuente == 'acerca':  
        velocidad_fuente_con_signo = -velocidad_fuente  
    else:  
        velocidad_fuente_con_signo = velocidad_fuente  
    # Fórmula Doppler general  
    frecuencia_observador = frecuencia_fuente * ((velocidad_sonido + velocidad_observador_con_signo) / (velocidad_sonido + velocidad_fuente_con_signo))  
    else:  
        print("Opción de escenario no válida. Por favor, intente de nuevo.")  
        return  
    # Mostrar resultado  
    print("La frecuencia percibida por el observador es: Hz", frecuencia_observador)  
    print("¡Gracias por usar la calculadora del Efecto Doppler!")  
efecto_doppler()
```

2) EJEMPLO

Calculadora del Efecto Doppler: calculamos la frecuencia percibida por un observador

Ingrese la frecuencia de la fuente (Hz): 400

Ingrese la velocidad del sonido en el medio (m/s): 343

Escenarios posibles

1) Fuente fija, observador en movimiento

2) Observador fijo, fuente en movimiento

3) Ambos en movimiento

Seleccione el número del escenario: 3

Ingrese la velocidad del observador (m/s): 20

¿El observador se ACERCA o se ALEJA de la fuente? (acerca/aleja): acerca

Ingrese la velocidad de la fuente (m/s): 10

¿La fuente se ACERCA o se ALEJA del observador? (acerca/aleja): aleja

La frecuencia percibida por el observador es: 411.3314447592068 Hz

¡Gracias por usar la calculadora del Efecto Doppler!

BIBLIOGRAFIA

RESNICK, R. HALLIDAY, D y KRANE, K. (2004) Física, Vol. I y II. Compañía Editorial Continental: México.

SERWAY, R. (1999) Física, tomo I y II. McGraw-Hill: México

SERWAY, R. y JEWETT, J. (2008) Física para ciencia e ingeniería. Vol 1 y 2. Cengage Learning: México.