TALLER DE DESARROLLO DE APLICACIONES . Práctica 1.

Cardona Núñez Mirna Itzell Xochiquetzal
Juan Carlos Garcia Acosta
Pérez Avila David Emmanuel

---En matemática y computación, el método de Euler, es un procedimiento de integración numérica para resolver ecuaciones diferenciales ordinarias (EDO) a partir de un valor inicial dado.

I. INTRODUCCIÓN

Los primeros en proponer la metodología de predicción fueron Euler e indirectamente Newton, donde gracias a estas metodologías lograron predecir la trayectoria de un cometa que pasaba en ese momento por encima de la tierra.
Un cambio pequeño en una cantidad está definida por el límite:

$$\frac{dy}{dt} = \lim_{\Delta t \to 0} \frac{y(t + \Delta t) - y(t)}{\Delta t}$$

Euler proponía que la mayoría de los fenómenos temporales dependían aproximadamente de tiempos pasados hasta cierto punto. Expresándose así matemáticamente:

$$\frac{dy}{dt} = f(y, t)$$

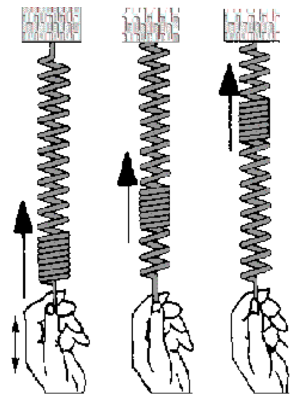
Esto quiere decir que el cambio de un fenómeno es función de algo que sucedió con el mismo en el pasado.

$$f(y, t) = \lim_{\Delta t \to 0} \frac{y(t + \Delta t) - y(t)}{\Delta t}$$

$$y(t + \Delta t) = \Delta t f(y, t) + y(t).\ \textit{Método de Euler}$$

En el método de Euler podemos conocer tiempos futuros a partir del estado actual y fuerzas que actúan en nuestro fenómeno de estudio.

Dentro esta pràctica se desea poder predecir el tiempo y trayectoria de una persona a la cual jalan desde abajo atado a una cuerda (resorte) y soltaran.



Para poder realizar esta predicciòn necesitamos utilizar la fòrmula fisica:

$$\sum F = ma$$

$$m\ddot{x} = -kx + mg$$

Y si utilizamos el mètodo de Euler para predecir la trayectoria.

$$x(t + 2\Delta t) = -x(t)\left[1 + \frac{k\Delta t^2}{m}\right] + 2x(t + \Delta t) + \Delta t^2 g$$

El sistema debe de ser capaz de determinar si en algún momento nuestro usuario del bungee llega a

pasar su distancia X0(distancia inicial), si esto sucede se estrellaria contra el suelo.
Para determinar si esto es posible se necesita saber de antemano los siguientes datos: la masa de la persona, constante del bungee jump, la constante de la gravedad y la la distancia inicial del movimiento.

II. ANÁLISIS

Se planea dividir nuestro código en módulos que nos permitan tener una mejor organización del mismo. En este caso crearemos 3 módulos. Uno principal,uno para meter la información a los archivos y uno de ecuaciones. Cada uno con sus respectivos .h y .c

El módulo principal será la vista, en el que se coloque un pequeño menú donde el usuario podrá insertar valores de la ecuación como la masa de la persona, la constante de elasticidad del bungee jump, el intervalo de tiempo y la duración del juego.

El módulo files.c se utilizará para meter la información obtenida por el módulo equation.c hacia los archivos, donde se enviará la información de los valores de la posición de la persona en el bungee jump y de los datos de tiempo de ejecución del programa.

El módulo equation.c se utilizará para como el motor donde se harán los cálculos de la ecuación de Euler para obtener los datos de la posición de la persona en el intervalo que el usuario pida.

III. PSEUDOCÓDIGO

main.c

```
void main{
FILE * arch1,arch2;
int tiempo = user input
int k = user input
int masa = user input
int cont //contador de ciclos
clock_t start, stop;
double cpu_tiempo, resultado;
arch1 = fopen("tiempoCPU.CSV", "w")
arch2 = fopen("tiempoRES.csv", "w")

for(cont = 0; cont <= tiempo; cont ++){
start = now()
resultado = equ(k,masa,tiempo,cont)
stop = now()
files
cpu_tiempo = stop-start;
imprimir(cpu_tiempo, cont, resultado)
resultado = 0
}end for
fclose(arch1)
fclose(arch2)
}end main
```

equation.c

equ(tiempo, k, masa,cont){
double res, eq1, eq2, eq3
m = masa, Δ t = tiempo, t = cont,g = 9.81
eq1 = $-x(t)[\frac{k\Delta t^2}{m} + 1]$
eq2 = $2x(t+\Delta t)$
eq3 = $\Delta t^2 g$
res = eq1+eq2+eq3;

return res;
}end equ

files.c

```
imprimir(cpu_tiempo,cont,resultado,arch1,arch2){
fprintf( arch1, cont \t resultado \n);
fprintf( arch2, cont \t cpu_tiempo \n);


}end imprimir
```

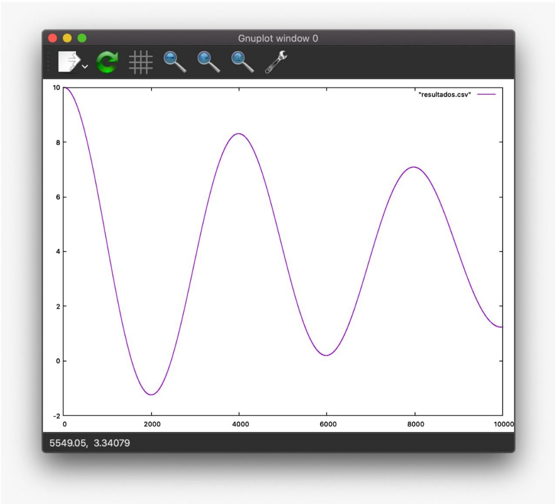
IV. RESULTADOS

Analizando las respuestas (gráficas) de nuestro sistema nos dimos cuenta que nunca se estabilizan, es decir que se mantienen oscilando infinitamente. Esto por supuesto no es una respuesta que se espera en un bungee, sin embargo nos dimos cuenta que esto se debe a que en nuestra ecuación no estamos considerando la resistencia natural del proceso, ya sea por el aire u otros aspectos, para solucionarlo y obtener una respuesta estable del sistema fue necesario agregar una constante al final de las ecuaciones:

```
res=(-xt * (((K * (Dt * Dt))/masa)+1-((BT*DT)/masa)));
```

y de `eq2=(2-(BT*DT)/masa)*xt;`

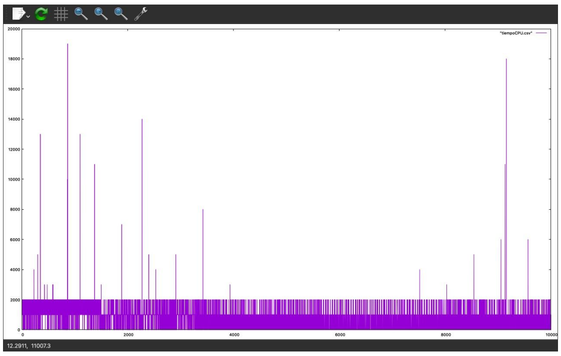
Una vez obteniendo una respuesta estable para nuestro sistema procedemos a analizar nuestros resultados.
Para un ejemplo de un tiempo de 10s, peso de 60kg, y la constante de bungee de 150 N/m nuestra respuesta nos queda de la siguiente manera:



Podemos notar que en ningún momento nuestro usuario (persona atada al bungee) choca o traspasa el suelo, por lo que es una buena manera de verificar que el bungee con estos parámetros será seguro.

Sin embargo también al jugar con nuestros parámetros nos damos cuenta que entre más pequeño sea la constante de nuestro bungee el sistema tiende a tener oscilaciones más cortas y frecuentes (su frecuencia aumenta) lo cual tampoco es deseable pues nuestra persona en el bungee se mareara demasiado y puede ser más peligroso, caso contrario pasa con una constante más grande ya que el sistema tendrá oscilaciones más largas y si bien no es malo tampoco es lo más deseable.

Por su parte en el lado de la medida del tiempo de cuanto se tardo en calcular la respuesta al sistema quedó de la siguiente manera:



Podemos darnos cuenta de que al ser programa que necesita hacer cientos de cálculos al momento nuestra gráfica tiene una marcada actividad en la parte inferior, esto se debe a que al ser tantos cálculos la gráfica se satura de información, si pudiéramos verla en una escala menor podemos darnos cuenta en qué procesos son los que le costaron más trabajo, en donde tardó más tiempo para poder procesarlos.

V. REFERENCIAS

Cesar A. (2020). Taller de Desarrollo de Aplicaciones Práctica 1.. Septiembre 24, 2020, Universidad Iberoamericana.

colaboradores de Wikipedia. (2020a, mayo 29). *Método de Euler*. Wikipedia, la enciclopedia libre. https://es.wikipedia.org/wiki/M%C3%A9todo_de_Euler#:%7E:text=El%20m%C3%A9todo%20de%20Euler%20es%20un%20m%C3%A9todo%20de%20primer%20orden,para%20construir%20m%C3%A9todos%20m%C3%A1s%20complejos.