

## MÉTODOS DE SIMULACIÓN – FÍSICA Taller 1, Ejercicios 1 y 2

Nota: la presentación en .pdf del taller debe contener las gráficas y respuestas que se solicitan, además de información adicional que enriquezca el problema (pertinencia, aplicaciones, conclusiones, etc). El envío debe ir acompañado de los programas .cpp o Python que se solicitan.

### Ejercicio 1: MODELO SIR

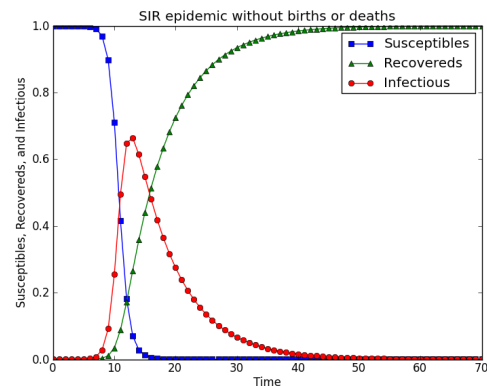
Uno de los modelos más sencillos para la evolución de una epidemia es el modelo SIR, propuesto en 1927 por W. O. Kermack & A. G. McKendrick que se utiliza para simular enfermedades en las que una persona infectada, una vez recuperada (o muerta) no se puede volver a infectar. Considere una población con un número constante de personas donde cada una puede estar *susceptible*, *infectada* (*casos activos*) o *retirada*. Sean  $s$ ,  $i$ ,  $r$  las fracciones de la población que están susceptibles, infectadas o retiradas, respectivamente ( $s + i + r = 1$ ). Asumamos que, a cada momento, la cantidad de susceptibles que se infecta por unidad de tiempo es proporcional a la cantidad de susceptibles multiplicada por la cantidad de infectados,  $\beta si$ , donde  $\beta$  es una constante que nos dice qué tan contagiosa es la enfermedad. Por el contrario, la cantidad de infectados que dejan de serlo por unidad de tiempo es solamente proporcional al número de infectados,  $\gamma i$ , donde  $1/\gamma$  da el tiempo característico de recuperación. Las ecuaciones diferenciales acopladas que rigen el comportamiento son, por lo tanto:

$$\frac{ds}{dt} = -\beta si$$

$$\frac{di}{dt} = \beta si - \gamma i$$

$$\frac{dr}{dt} = \gamma i$$

, de las cuales sólo las dos primeras son independientes.



a) Implemente la simulación del modelo utilizando Runge-Kutta para las dos primeras ecuaciones acopladas, y grafique la evolución temporal de las tres cantidades  $s(t)$ ,  $i(t)$  y  $r(t)$  para  $s(0) = 0.999$ ,  $i(0) = 0.001$ ,  $\beta = 0.35$ ,  $\gamma = 0.08$ .

c) El archivo [CoronavirusColombiaBogota.xmls](#) contiene los datos de los casos reportados de Covid-19 para Colombia y para Bogotá.

- Escoja un grupo de datos (Colombia o Bogotá) y utilice los datos del inicio de la pandemia para deducir una tasa de crecimiento diario  $a = i(t+1)/i(t)$  e identifique un día en el que este valor ya esté estable. Identifique ese día como el momento inicial de la pandemia  $t = 0$ . Calcule los valores de iniciales como  $r = 0$ ,  $s = 1 - i$  e  $i = I/N$ , utilizando el número de casos activos  $I$  en ese momento y una población de  $N=45,000,000$  para Colombia o  $N=8,000,000$  para Bogotá.
- Luego, considere el siguiente procedimiento, que se puede considerar como si fuera una función  $f(\gamma)$ : Dado un valor de  $\gamma$ , calcule  $\beta$  a partir de  $a$ . Con estos valores de  $\beta$  y  $\gamma$ , corra la simulación del modelo SIR. Como resultado de la simulación, calcule el número de días  $t_{\text{simulado}}$  en el que se presenta el pico en la simulación, y defina la función  $f(\gamma) = t_{\text{simulado}} - t_{\text{datos}}$ , donde  $t_{\text{datos}}$  es el día en el que ocurre el pico, contado desde el momento inicial.
- Finalmente, utilice el método de la bisección de esta "función" para identificar el valor de  $\gamma$  que predice correctamente el momento de aparición del pico ( $f(\gamma) = 0$ ).

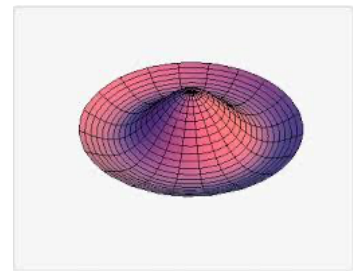
El envío debe contener:

- El programa (.cpp o Python) que resuelve el punto a), junto con la gráfica de evolución de las tres cantidades.
- El desarrollo del punto b).
- El programa (.cpp o Python) que resuelve el punto c), junto con los valores hallados para  $\alpha$ ,  $\beta$  y  $\gamma$ , y la curva de los datos experimentales y de la simulación que mejor ajusta esos datos.

## Ejercicio 2: LOS MODOS DE VIBRACION DE UN TAMBOR

Considere la membrana circular de un tambor de radio  $a = 1$ , e imagine que estamos interesados en los modos normales  $R(r)$  que no dependen de la coordenada angular  $\theta$ , como la que se muestra en la figura. En este caso, la función  $R(r)$  cumple la ecuación diferencial

$$r^2 \frac{d^2 R}{dr^2} + r \frac{dR}{dr} + \lambda^2 r^2 R(r) = 0 \quad ,$$



que resulta ser una ecuación de Bessel con parámetro  $\alpha = 0$ , con condiciones de frontera  $\frac{dR}{dr}_{r=0} = 0$  y  $R(a) = 0$ .

a) Construya un programa que resuelva esta ecuación diferencial por Runge-Kutta de 4º orden para  $\lambda = 1$ ,  $R(0) = 1$  y  $\frac{dR}{dr}_{r=0} = 0$ , y grafique  $R(r)$  en el intervalo  $r \in [0.01, 10.0]$ .

b) Defina la función  $f(\lambda) = R(r = 1; \lambda)$  (donde hemos hecho explícito el hecho de que  $R(r)$  también es función del parámetro  $\lambda$ ), gráfiquela para  $\lambda \in [0.1, 15.0]$  e identifique aproximadamente dónde están los ceros de la función en ese intervalo (es decir, los valores aproximados de  $\lambda$  para los cuales  $f(\lambda) = 0$ )

c) Utilizando el método de hallar ceros por bisección, precise los valores de  $\lambda$  para los cuales se logra que  $R(r = a) = 0$  (con  $a = 1$ ), es decir para los que se cumplen las condiciones de frontera del tambor, y grafique las funciones  $R(r)$  correspondientes en el intervalo  $r \in [0.01, 1.0]$ . Estos son los modos normales que deseábamos encontrar.

d) Haga un programa que grafique las funciones de Bessel que son solución teórica de este problema, halle su ceros por bisección y compárelos con los valores que encontró en el punto c.

El envío se debe hacer en un solo correo electrónico, que contenga:

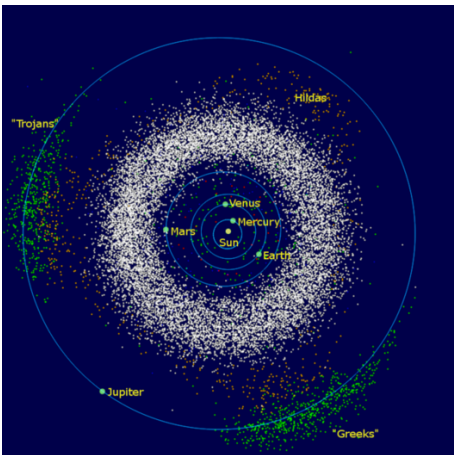
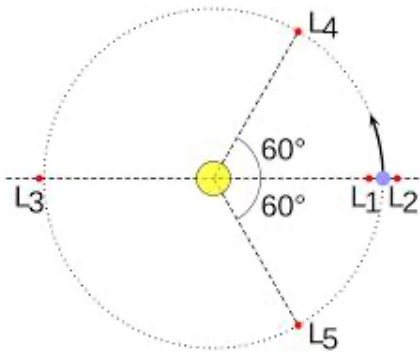
- El programa .cpp o Python que resuelve el punto a), junto con la gráfica de  $R(r)$ .
- El programa .cpp o Python que resuelve el punto b), junto con la gráfica de  $f(\lambda)$ .
- El programa .cpp o Python que resuelve el punto c), junto con la gráfica de los modos normales y los valores de  $\lambda$  para cada uno de ellos.
- El programa .cpp o Python del punto d), junto con una tabla que compare los valores de  $\lambda$  hallados por los dos métodos.

# MÉTODOS DE SIMULACIÓN – FÍSICA Taller 1, Ejercicio 3

## PLANETAS TROYANOS

- a) Hemos simulado en clase el movimiento de dos cuerpos celestes. Considere que uno de ellos el Júpiter y el otro es El Sol, cuya masa es  $m_0 = 1047$  veces la masa de Júpiter ( $m_1 = 1$ ). Asuma que la distancia entre Júpiter y El Sol es  $r = 1000$  y coloque las condiciones iniciales para que el movimiento de los dos cuerpos sea girar en círculos alrededor de su centro de masa con velocidad angular  $\omega$ . Cuadre el paso de tiempo  $\Delta t$  para que un total de 20 órbitas cierren correctamente.
- b) Ahora, en vez de imprimir las coordenadas  $x,y$  de Júpiter, imprima las coordenadas  $x_{\text{rotado}}, y_{\text{rotado}}$  en un sistema que rota con Júpiter, de tal manera que tanto El Sol como Júpiter se ven quietos.

Si desde este sistema rotante se estudia el movimiento de un tercer cuerpo cualquiera, se observa sobre él el efecto de cuatro fuerzas: la atracción de El Sol, la atracción de Júpiter, la fuerza centrífuga y la fuerza de Coriolis, pero ésta última no afecta a cuerpos que se vean en reposo desde el sistema rotante. Lagrange (en 1772) calculó los puntos en los que las tres fuerzas restantes se equilibran, que se conocen como **puntos de libración**. De ellos, sólo dos son puntos estables (L4 y L5), y se encuentran sobre la órbita circular de Júpiter,  $60^\circ$  en adelante y  $60^\circ$  en atraso (ver Figura). Un tercer planeta colocado en reposo en estos puntos (es decir, rotando con la misma órbita de Júpiter pero  $60^\circ$  en adelante o en atraso), permanecería en reposo. De hecho, existe una infinidad de planetas, llamados troyanos, que orbitan alrededor del sol cerca de estos dos puntos.

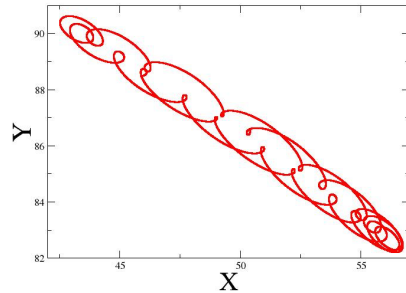


588 Achilles
624 Hektor
659 Nestor
911 Agamemnon
1143 Odysseus
1404 Ajax
1437 Diomedes
1583 Antilochus
1647 Menelaus
1749 Telamon
1868 Thersites

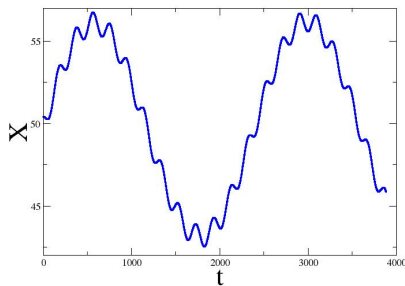
617 Patroclus
884 Priamus
1172 Áneas
1173 Anchises
1208 Troilus
1867 Deiphobus
1870 Glaukos
1871 Astyanax
1872 Helenos
1873 Agenor
2207 Antenor

c) Coloque un tercer planeta de masa  $m_3 = 0.005$  en el punto de libración L4 girando con la misma rapidez que Júpiter, y compruebe que permanece aproximadamente quieto por 20 órbitas.

d) Perturbe un poco la posición o la velocidad inicial del planeta troyano (en unas cuantas partes por mil) y observe que el planeta dibuja en el sistema rotante una trayectoria en espiral, como la que muestra la figura. Esta espiral corresponde, aproximadamente, a una elipse moviéndose sobre otra elipse, como en los epiciclos de Ptolomeo. Los periodos de estos dos movimientos elípticos superpuestos se pueden calcular por teoría clásica de perturbaciones, y resultan ser:



$$T_1 = \frac{T}{\sqrt{\frac{27}{4} \frac{m_1}{m_0}}} , \quad T_2 = \frac{T}{1 - \frac{27}{8} \frac{m_1}{m_0}} ,$$



donde  $T = 2\pi/\omega$  es el período de rotación del sistema Sol-Júpiter.

e) Grafique la posición  $x$  del planeta troyano en el sistema rotante en función del tiempo y mida aproximadamente los dos periodos. Compare los valores obtenidos con los predichos por la teoría clásica de perturbaciones.

f) Sabiendo que en las unidades de trabajo de este problema la masa de Júpiter ( $1.898 \times 10^{27}$  kg) es 1, la distancia media de Júpiter al Sol (778'412,026 km) es 1000, y la constante gravitacional  $G$  ( $6.67408(31) \times 10^{-11} \text{ m}^3 \cdot \text{kg}^{-1} \cdot \text{s}^{-2}$ ) es 1, halle a cuántos segundos corresponde una unidad de tiempo, y dé los periodos del punto d) en años.

### Para la entrega

El envío (.pdf de la presentación y programas .cpp o Python) debe contener:

- El programa que resuelve el punto a), es decir que simula el sistema Sol-Júpiter en el sistema sin rotar, junto con la gráfica de las 20 órbitas.
- El programa que resuelve el punto b), es decir que simula el sistema Sol-Júpiter en el sistema rotado, junto con la gráfica de las órbitas.
- El programa que resuelve el punto c), es decir que muestra los tres cuerpos en el sistema rotado y comprueba que el planeta troyano prácticamente permanece inmóvil en dicho sistema. junto con la gráfica correspondiente.
- El programa que resuelve el punto d), es decir que perturba la posición o la velocidad del planeta troyano y grafica su órbita espiral en el sistema rotado, junto con la gráfica de esa órbita espiral.
- El programa que resuelve el punto e), es decir que muestra la componente  $x$  de la trayectoria espiral del planeta troyano en el sistema rotado, en función del tiempo, junto con la gráfica correspondiente.
- Además, el texto del correo debe tener los dos períodos, tanto con sus valores teóricos como con los obtenidos de la simulación, además de la deducción del punto f).

### Referencias

- [1] S.W. McCuskey and S.W. Reading , *Introduction to Celestial Mechanics*, (Addison Wesley, 1963).

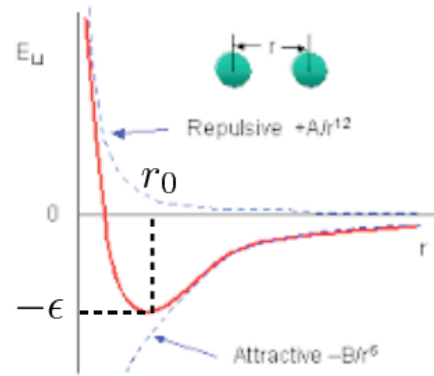
## MÉTODOS DE SIMULACIÓN – FÍSICA Taller 1, Ejercicio 4

### GAS DE LENNARD-JONES

El potencial de Lennard-Jones,

$$V(r) = \varepsilon \left[ \left( \frac{r_0}{r} \right)^{12} - 2 \left( \frac{r_0}{r} \right)^6 \right],$$

es un modelo aproximado de la interacción entre moléculas no polares que combina una fuerza repulsiva de exclusión de Pauli (el término a la potencia 12) con una fuerza atractiva debida a fuerzas de van der Waals (el término a la potencia 6). El potencial presenta un valor mínimo  $-\varepsilon$  a una distancia  $r_0$ , que es la distancia de equilibrio (ver Figura). La fuerza derivada de este potencial,



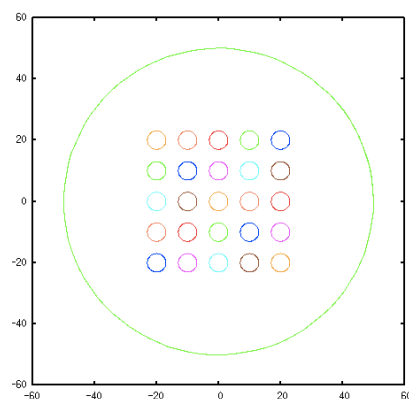
$$F(r) = \frac{12\varepsilon}{r} \left[ \left( \frac{r_0}{r} \right)^{12} - \left( \frac{r_0}{r} \right)^6 \right],$$

es una fuerza restitutiva alrededor de  $r_0$ , que corresponde más o menos a la de un resorte que es difícil de comprimir, pero fácil de estirar.

- 1) Construya un programa que simule el movimiento de una partícula bidimensional de masa  $m$  bajo el influjo de una fuerza de Lennard-Jones, implementada como una fuerza central. Utilice  $m = 1.0$ ,  $R = 3.0$ ,  $\varepsilon = 1.0$  y  $r_0 = 10$ . Como condición inicial coloque la partícula en  $x = 10$ ,  $y = 0$ ,  $V_x = V_0$  y  $V_y = 0$ , con  $V_0 = \sqrt{k_B T / m}$ , donde  $k_B T = 1.0$  es la temperatura.
  - a. Observe cómo oscila.
  - b. Grafique  $x$  vs  $t$  para  $0 < t < 100$ , y cuadre el paso  $\Delta t$  para que la energía no crezca (esto se puede comprobar verificando que la curva no cambia si se hace  $\Delta t$  diez veces más pequeño).

De esta primera parte, presente:

- El programa .cpp o Phthon
  - La gráfica del punto 1b
- 2) Ahora el objetivo es construir un gas de partículas que interactúan por Lennard-Jones. Para eso, modifique el programa Gas2D visto en clase de la siguiente manera:
    - a. Modifique la fuerza entre moléculas para que sea la de Lennard-Jones, con los mismos parámetros del punto anterior.
    - b. Como condición inicial, coloque un cuadrado de  $5 \times 5$  partículas en las posiciones  $x = -20, -10, 0, 10, 20$  y  $y = -20, -$



- 10,0,10,20, con magnitud de velocidad inicial  $V_0 = \sqrt{k_B T / m}$ , como en el caso anterior, y dirección de la velocidad  $\theta$  escogida al azar entre 0 y  $2\pi$ .
- Quite los granos que hacen las veces de paredes. En su lugar, dibuje un círculo de radio  $R_{\text{pared}}=50$  centrado en el origen y añada una fuerza global de rebote ejercida por esa pared circular de la siguiente manera: si el grano de radio  $R$  penetra en la pared una distancia  $s$ , sentirá una fuerza de magnitud  $F = Ks$  en dirección hacia el origen, con  $K=10000$ .
  - Fije el paso de tiempo a  $\Delta t = 5 \times 10^{-4}$ , el tiempo máximo de simulación en  $t_{\text{max}}=100$  y dibuje en la animación un total de 2000 cuadros.
  - Vea qué pasa si  $k_B T = 0.05$  (un sólido),  $k_B T = 0.5$  (un líquido), y  $k_B T = 10.0$  (un gas).

De esta segunda parte, presente:

- El programa .cpp o Python
  - Un .gif animado o un .mp4 de la simulación para cada uno de los estados: sólido, líquido y gas, a lo largo de un tiempo de simulación  $t_{\text{max}}=100$ .
  - Describa, además, lo que observa.
- 3) Ahora vamos a construir una función que mida de manera aproximada el tamaño ocupado por los  $N=25$  granos. Para ello, utilizaremos el radio de giro, que (como todas las masas son iguales) es dado por

$$R_{\text{giro}} = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} \|\vec{r}_i - \vec{r}_{\text{cm}}\|^2}$$

- Como primera parte de la función, calcule el vector posición  $\vec{r}_{\text{cm}}$  del centro de masa de todos los granos, como

$$\vec{r}_{\text{cm}} = \frac{1}{N} \sum_{i=0}^{N-1} \vec{r}_i$$

- Complete la función para que calcule y devuelva el valor  $R_{\text{giro}}$ .
- En el programa principal, deje de hacer la animación y en vez de ello imprima  $R_{\text{giro}}$  en función de  $t$ . Grafique en una sola figura  $R_{\text{giro}}$  en función de  $t$  para los casos de sólido, líquido y gas ( $0 \leq t \leq 100$ ).
- Calcule el valor promedio de  $R_{\text{giro}}$  desde  $t=20$  hasta  $t=100$  para los casos de sólido, líquido y gas. ¿Qué concluye de ello?

De esta tercera parte, presente:

- El programa .cpp o Python
- La gráfica de  $R_{\text{giro}}$  vs  $t$  para sólido, líquido y gas del punto c)
- Presente, además los valores promedio de  $R_{\text{giro}}$  para sólido, líquido y gas, y lo que concluye al analizar estos resultados.

## Referencias

- [1] [https://es.wikipedia.org/wiki/Potencial\\_de\\_Lennard-Jones](https://es.wikipedia.org/wiki/Potencial_de_Lennard-Jones)

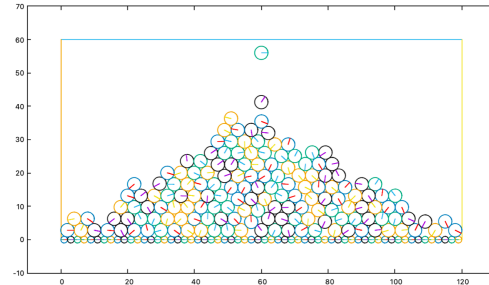


## MÉTODOS DE SIMULACIÓN – FÍSICA Taller 1, Ejercicio 5

### PILA DE ARENA

Una pila de arena se puede construir arrojando granos uno a uno sobre una superficie, hasta formar un cono. En teoría, el ángulo  $\theta$  que forma la superficie del cono con la horizontal – que se conoce con el nombre de *ángulo de reposo* – debe ser tal que su tangente sea igual al coeficiente de fricción estático  $\mu$ ,

$$\mu = \tan \theta . \quad (1)$$



Para comprobarlo,

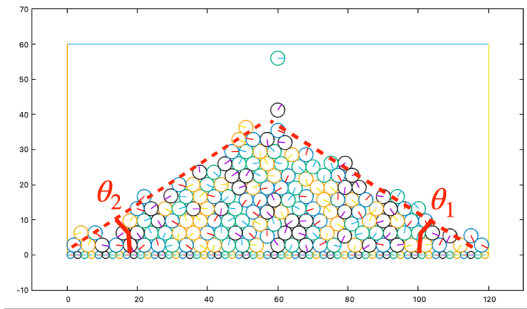
- a) Construya un programa basado en el código `Granos2D_2024-I.cpp` que genere una pila de arena, soltando grano por grano desde cierta altura. Para ello, utilice un espacio de simulación de, por ejemplo,  $L_x=160$  y  $L_y=60$  y represente el suelo como un conjunto de  $N_s=80$  esferas de radio  $R_s=L_x/(2*N_s)=1$  que no se mueven ni interactúan entre sí ni con las otras tres paredes, como muestra la figura (si lo ve conveniente, puede ampliar  $L_x$  y/o  $N$  aún más). Haga que todos los  $N = 200$  granos inicien con una posición  $x_0=L_x/2$ ,  $y_0=L_y-2R_0$ , con  $R_0=2$ , velocidad  $V_x=V_y=0$ , ángulo  $\theta_0=0$  y una velocidad angular  $\omega$  que se escoge aleatoriamente en el intervalo  $\omega \in [-\theta_{\text{Max}}, \theta_{\text{Max}}]$ , pero vaya soltándolos uno por uno, dejando evolucionar el sistema un tiempo  $t_{\text{max}} = 5\sqrt{L_y/g}$  antes de soltar el siguiente. Una forma de logarlo es iniciar todos los granos antes de comenzar la simulación y definir una variable entera  $N_{\text{live}} = 0, 1, 2, \dots$  que indica cuántos granos están vivos. Para la simulación puede tomar como valores de referencia los siguientes:

```
//----- declarar constantes ---  
const double K=1.0e4;  
const double Lx=160, Ly=60;  
const int N=200, Ns=80, Ntot=N+Ns+3;  
const double g=9.8, Gamma=150, Kcundall=500, mu=0.4;
```

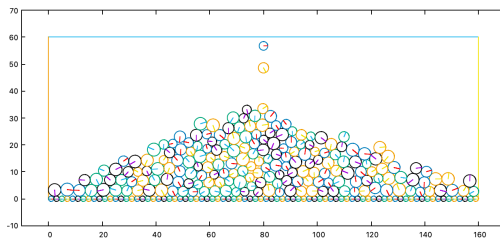
y como variables iniciales en el `main()` las siguientes:

```
int main(void){  
    Cuerpo Molecula[Ntot];  
    Crandom Ran64(1);  
    Colisionador Hertz;  
    Crandom ran64(1);  
    double m0=1, R0=2, kT=10, V0=sqrt(2*kT/m0);  
    int i,Nlive;  
    double cuadros=5,t,tdibujo,dt=1e-3  
        |,tmax=cuadros*sqrt(Ly/g),tcuadro=tmax/(10*cuadros);  
    double Omega0,OmegaMax=8.0;  
    double Rpared=100*Lx, Mpared=100*m0;  
    double Rs=Lx/(2*Ns);
```

- b) Una vez se complete la pila, ajuste dos líneas a sus perfiles, como muestra la figura, y mida su ángulo de inclinación, que corresponde al ángulo de reposo  $\theta$ , y compruebe si se cumple la relación Ec. (1). Si lo desea, tome  $\theta = (\theta_1 + \theta_2)/2$ , con  $\theta_1$  y  $\theta_2$  los ángulos medidos a derecha e izquierda, como se muestra en la figura.



- c) Ahora repita todo el proceso anterior, pero haciendo que cada grano que se suelta tenga un radio escogido al azar entre 1.6 y 2.4 ( $R_0 \in [1.6, 2.4]$ ). Observe si en este caso el ángulo  $\theta = (\theta_1 + \theta_2)/2$  se acerca más o no al valor predicho por la Ec. (1).



### Para la entrega

El envío (.pdf de la presentación y programas .cpp o Python) debe contener:

- El programa que implementa la simulación del punto a).
- El programa que implementa la simulación del punto c)
- Las gráficas de las dos pilas de arena, con los ajustes respectivos, y el cálculo de error porcentual respecto del dato teórico.

### Referencias

[1] Angle of repose for two-dimensional particle aggregates from particle-size and -shape Salah A. M. El Shourbagy & Hans-Georg Matuttis, University of Electro-Communications, Department of Mechanical Engineering and Intelligent Systems, Tokyo, Japan



## MÉTODOS DE SIMULACIÓN – FÍSICA Taller 1, Ejercicio 6

### RELOJ DE ARENA EN YADE

El flujo de masa de un material granular cuando este fluye a través de un orificio, como en un reloj de arena, está dada por la relación de Hagen-Beverloo [1]:

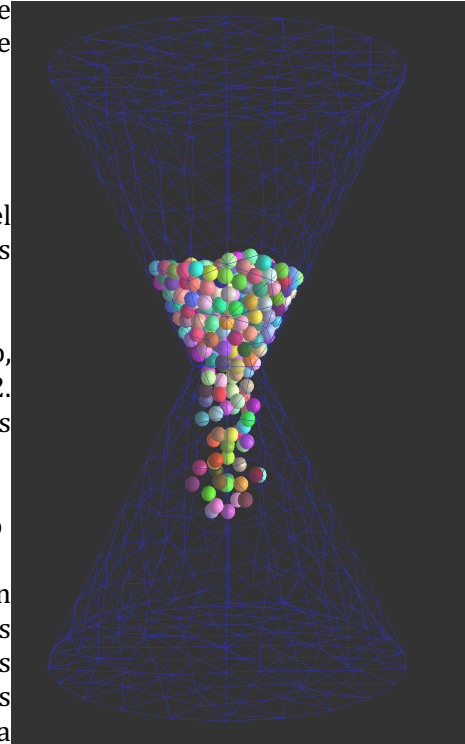
$$W = 35 \rho_B \sqrt{g} (D_0 - 1.4 d)^{2.5}$$

Donde  $\rho_b$  es la densidad bulk del material,  $g$  la gravedad,  $D_0$  el diámetro del orificio y  $d$  el diámetro de los granos. Si todas las unidades están en SI,  $W$  tendrá unidades de Kg/min.

- a) Utilizando el programa gmsht y el archivo hourglass.geo, identifique a que corresponden las variables  $r_0$ ,  $r_1$ ,  $r_2$ ,  $h_1$  y  $h_2$ . Cree un reloj de  $h_1 = h_2 = 20$  cm y  $r_1 = r_2 = 13$  cm. Cree varios relojes con un radio de apertura de 2.5, 2.0 y 1.5 cm.

Gmsh -3 -format mesh -optimize\_netgen -o input.mesh output.geo

- b) Cree una simulación donde pueda llenar el reloj de arena con esferas de 0.55 cm de radio. Cree además algo que permita que las esferas se asienten sin caer por el hueco del reloj. Asuma que las paredes del reloj están hechas de vidrio boro-silicato y los granos de silica ( $\text{SiO}_2$ ) y asigne las respectivas propiedades materiales a los elementos de la simulación. Utilice el modelo de fuerzas de Hertz-Mindlin para definir las interacciones entre los elementos de la simulación. Utilice un coeficiente de amortiguamiento de 0.93 para ambos materiales.
- c) Investigue en qué consiste el modelo de Hertz-Mindlin y como la fuerza depende de las propiedades del material. Además, averigue cuales son los dos modelos de amortiguamiento implementados en el modelo de Hertz-Mindlin de Yade, quién los propuso y como difieren el uno del otro.
- d) Idee un mecanismo para eliminar el bloqueo del reloj una vez los granos estén en equilibrio. Defina un criterio para saber cuándo los granos están en equilibrio.
- e) Cree una función que pueda medir el flujo de partículas por el centro del reloj.
- f) Para poder medir la densidad Bulk del material es necesario conocer el factor de empaquetamiento de los granos en la simulación. Yade tiene una función capaz de medir este factor. Se llama `yade._utils.porosity()`. Averigue cómo funciona esta función. El factor de empaquetamiento (para esferas de radios iguales) cambia con el radio? La densidad Bulk la puede calcular como la densidad del material por el factor de empaquetamiento.
- g) Utilice yade-batch para correr las simulación con todos los distintos relojes de arena creados en el apartado c). Por cada reloj de arena, repita la simulación con 400 esferas de 0.55, 0.50 y 0.45 cm de radio. Si puede utilizar más esferas, mejor.



- h) Verifique que los datos obtenidos en la simulación se ajustan al modelo teórico del flujo de arena en el reloj.

### Para la entrega

El envío (.pdf de la presentación y programas .cpp o Python) debe contener:

- a) El PDF con el desarrollo teórico de los puntos c) y f)
- b) El programa que implementa la simulación
- c) Presentación de flujo en cada uno de los relojes y la comparación con la teoría.

### Referencias

[1] W.A. Beverloo, H.A. Leniger, J. van de Velde, The flow of granular solids through orifices, Chemical Engineering Science, Volume 15, Issues 3–4, 1961, Pages 260-269, [https://doi.org/10.1016/0009-2509\(61\)85030-6](https://doi.org/10.1016/0009-2509(61)85030-6).