

Informe de Métodos Numéricos

Tarea 2

Leonardo Leiva

1. Introducción

En este informe se mostrará la modelación del movimiento de una partícula sobre una plataforma que se mueve sinusoidalmente. La partícula choca inelásticamente con el suelo. Se analizará su comportamiento en una cantidad considerable de rebotes para ver comportamientos que tienden a ser sostenidos en el tiempo.

2. Caracterización del Problema

El problema consiste en una partícula de masa m que rebota contra el suelo, el cual es sometido a un forzaje sinusoidal de amplitud A y frecuencia ω . Para reducir la complejidad del problema se reducirá el movimiento de la partícula solo al eje Y (perpendicular al suelo). El choque que caracteriza la influencia del suelo sobre la partícula disipa energía (no es elástico), caracterizado por la fórmula:

$$v_p'(t^*) = (1 + \eta)v_s(t^*) - \eta v_p(t^*) \quad (1)$$

donde η es el coeficiente de restitución (toma valores de 0 a 1, donde 0 es completamente inelástico y 1 es el caso elástico), t^* es el momento del bote, v_p y v_s son las velocidades de la partícula y el suelo respectivamente antes del bote.

Para comenzar se considerará que la partícula estará inicialmente en contacto con el suelo y su velocidad será mayor que la de este. Para simplificar, se tomará el forzaje sin una fase (por lo tanto, se comportará proporcional a $\sin(\omega * t)$ donde t es el tiempo).

Es necesario definir los parámetros que se usarán. Por simplicidad, se tomará $A = 1$, $m = 1$ y $g = -1$. El uso de A y g tiene sentido y aplicaciones prácticas en métodos físicos experimentales al definir una aceleración normalizada $\Gamma = A\omega/g$ que es utilizada para trabajar medios granulares sometidos a un forzaje. El parámetro $y(0)$ también ha sido fijado, por lo que falta fijar $v(0)$, η y ω que serán escogidos más adelante.

3. Procedimiento: Búsqueda de la Posición y Velocidad del Choque n-ésimo

3.1. Las posiciones

Se plantea el problema de escribir un programa que, dada una posición y una velocidad (del choque n-ésimo) entregue la posición y velocidad del siguiente instante. Para ello hay que caracterizar el movimiento de la partícula y del suelo. Como se mencionó parcialmente, podemos escribir el movimiento del suelo como (2). Antes de mencionar el de la partícula, por facilidad, se ha decidido agregar un parámetro nuevo recibido y entregado por la función. Este corresponde al tiempo del choque n-ésimo. De esta manera, el movimiento de la partícula para el choque n-ésimo queda definido por caída libre de la forma (3).

$$y_s(t) = A \sin(\omega t) \quad (2)$$

$$y_p(t) = \frac{g}{2}(t - t_n)^2 + v_n(t - t_n) + y_n \quad (3)$$

Llamaremos también $y(t) = y_p(t) - y_s(t)$. Inicialmente no se había considerado el desplazamiento en t_n para cada bote. Este procedimiento está descrito en los programas "new.py", "new1.pyz" "new.p2" donde se implementaron las primeras tres partes de la tarea considerando que en cada caso se volvía al origen temporal. Haciendo un apropiado trato de estos datos se podrían haber obtenido los datos buscados, pero era más práctico implementar las ecuaciones (2) y (3).

El programa "n1.py" tiene la rutina para calcular los datos importantes del choque $n + 1$. Recibe los parámetros v_n , y_n y t_n y busca los ceros de la función $y_p(t) - y_s(t)$ mediante el método de la bisección incluido en "scipy". Para ello se considerarán los puntos $t_n + \epsilon$, $t_{n\max}$ y $t_{n\min}$, donde el primero correspondería a la raíz que conocemos (y con la cual definimos la parábola) más un número muy pequeño para que no nos arroje la misma raíz (queda pendiente verificar que tan pequeño sería el ϵ óptimo para este problema. Se escogió uno apropiado según los parámetros recorridos), el segundo punto es el vértice de la parábola que se puede obtener derivando e igualando a cero y el último corresponde a igualar $y_p(t')$ al mínimo valor que pueda tomar la intersección entre y_s e y_p , que corresponde a $-A$. Esto porque, a lo mas se pueden intersectar en ese punto, por lo tanto, la raíz necesariamente estará antes.

$$t_{n\max} = -v_n/g + t_n \quad (4)$$

$$t_{n\min} = (-v_n - \sqrt{v_n^2 - 2g(y_n + A)})/g + t_n \quad (5)$$

Es necesario mencionar también las velocidades que juegan un rol importante en este problema.

3.2. Las velocidades y la velocidad relativa

Definiremos las velocidades para la partícula como v_p y la del suelo como v_s , las cuales están dadas por las derivadas de las posiciones:

$$v_n = g(x - t_n) + v_n \quad (6)$$

$$v_s = A\omega \cos(\omega x) \quad (7)$$

En el desarrollo del problema se presenta la posibilidad de que la partícula adquiera una velocidad muy cercana a la del piso. Esto implicaría que la partícula se quedaría pegada al piso. En este sentido, si se piensa en que casos pasaría eso, sería necesario, en primera instancia, tener mayor libertad para escoger g , A y ω . Si imaginamos una partícula que inicialmente está en el piso y se comienza a hacer vibrar el piso pero la amplitud es muy pequeña. Esta situación resultará en que no se despegará del piso. De la misma manera, si se estuviera en un lugar que hay mucha aceleración de gravedad se necesitaría una amplitud muy grande para lograr mover la partícula. Por último, si tenemos una situación con características estándar (aceleración y amplitud no muy grandes), pero la frecuencia de oscilación del piso es muy baja, la partícula no alcanza a afectarse por esto y se moverá de la misma manera que el plano.

Teniendo en cuenta esta posibilidad, los parámetros podrían entregar una solución en la cual la partícula quedaría pegada. Se intenta implementar una opción en la cual, si la velocidad del plano es mayor que la partícula buscará un punto en el cual la aceleración del suelo sea 0, en donde la partícula (dado que adquirió velocidad con el suelo) pudiera salir despegada del piso.

Tiene errores en implementación ya que debería considerar la velocidad después del choque pero no se alcanzó a perfeccionar. Por otro lado, podría no solo buscar cuando la velocidad del plano sea mayor, ya que, por la ecuación (1), esto no debería pasar, si no que considerar una diferencia de velocidades muy pequeñas. En rigor, este efecto provocaría muchos rebotes infinitesimales, pero si se agrega la condición de que quede pegada al plano, se puede buscar el punto en el cual la partícula se despegaría mediante un diagrama de fuerzas igualando la normal a 0, y darle a la partícula la velocidad del plano en el punto de despegue, continuando el ciclo de los rebotes. La ecuación que permitiría obtener en tiempo en el que se despega sería:

$$t_d = \arcsen\left(\frac{g}{A\omega^2}\right)/\omega + t_n \quad (8)$$

Queda pendiente su posible implementación para modelar de mejor manera el caso de que una partícula. Por cierto, dependiendo de los parámetros g , A y ω , el despegue existiría en algunos caso, pero sería tal que la partícula sigue pegada al suelo.

3.3. Obtención de la Velocidad y Posición

Continuando lo que se mencionó anteriormente, luego de tener el punto cercano a la primera raíz, el vértice y la mínima raíz posible en el ciclo n -ésimo, se puede determinar mediante la multiplicación de $y(t_n + \epsilon) * y(t_n máx)$, ver su signo, si es negativo, el cambio de signo indica que hay una raíz ahí. Es necesario mejorar el programa en este caso, porque si el rebote ocurre muy cerca de la raíz original podría quedar fuera del rango ϵ , mientras que para valores muy pequeños, podría reconocer por segunda vez la misma raíz. Suponiendo que estos casos estuvieran cubiertos, basta con aplicar la función bisección dándole como

parámetro la función analítica $y(t)$ y los puntos mencionados anteriormente. Si la multiplicación no es negativa, indica que el cambio de signo ocurre entre el vértice y la mínima raíz posible, por lo que se repite el proceso para tal intervalo.

En el programa, paralelo a este cálculo está parte de la implementación del método que detecta cuando la partícula se queda pegada al suelo mencionado anteriormente. Como se dijo, queda mejorarlo y corregirlo, pero se dejaron ahí por la posibilidad de alcanzar a implementarlo antes de entregar la tarea.

Finalmente el programa devuelve la velocidad de la partícula de acuerdo al caso pertinente: si es choque, devuelve la velocidad con la fórmula, si fuera por quedarse pegado al suelo devuelve la velocidad del suelo en el punto de despegue, luego la posición de despegue y finalmente el tiempo en el que ocurre.

3.4. N de Relajación

Se usarán los parámetros $\eta = 0,15$, $\omega = 1,66$ que vienen designados por enunciado y $v_0 = 2$ porque se trabajó la mayor parte del tiempo con este parámetro inicial.

Para estimar el N_{Relax} se usó una iteración que tomaba los valores iniciales, calcula los parámetros para el siguiente choque con la función descrita anteriormente. Luego guarda los parámetros útiles en listas (velocidad, posición y tiempo). La lista con el número de iteraciones era solo para graficar. Luego se grafica el vector de velocidades con respecto al número del choque "n".

Se puede observar que cada vez la variación de velocidades se atenúa más en torno a $v = 1,9$ aproximadamente. Podemos estimar de forma no muy rigurosa que en $N_{Relax} = 60$. Se puede ver en la figura 1.

Luego se comenzó a variar el valor de ω desde 1.66 hasta 1.7 para ver el N_{Relax} en tales casos.

Los fenómenos observados destacables entre ω 1.66 y 1.7 son los que se muestran en las figuras 2, 3 y 4. Primero el n de relajación bastante grande (aunque se observa tendencia muy marcada desde bastante pronto). No es el caso de valores aproximadamente en 1.67 donde se podría decir que la atenuación ocurre a n mucho mas grandes que 200 (que fue el valor usado para la iteración). Si se sigue aumentando se puede observar que la velocidad se estabiliza en dos valores diferentes. Para valores un poco mayores que 1.67 (cuando ya se dispersan completamente ambas convergencias) existe un tiempo de relajación distinguible pero no muy grande (figura 3) hasta que practicamente desaparece en la figura 4 para $\omega = 1,7$.

Se tomará $N_{Relax} = 60$ para las partes siguientes como referencia para cubrir el caso de 1.66. Para los casos en los que no converja a un número único se considerará que se estabilizan en estos (usualmente menor a 60).

El programa que tiene el desarrollo de esta parte es "n2.py".

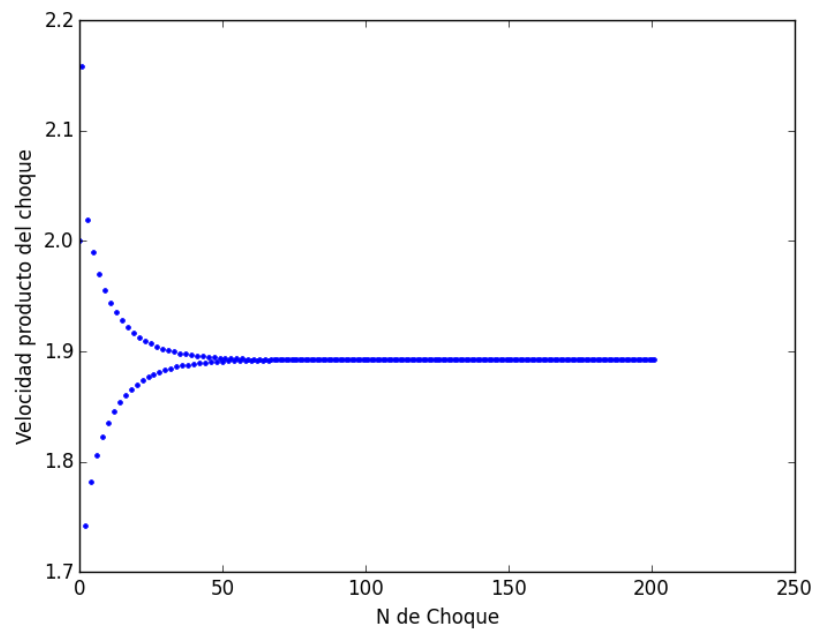


Figura 1: Relación entre la velocidad con la que rebota la partícula y el número del choque a $\omega = 1,66$

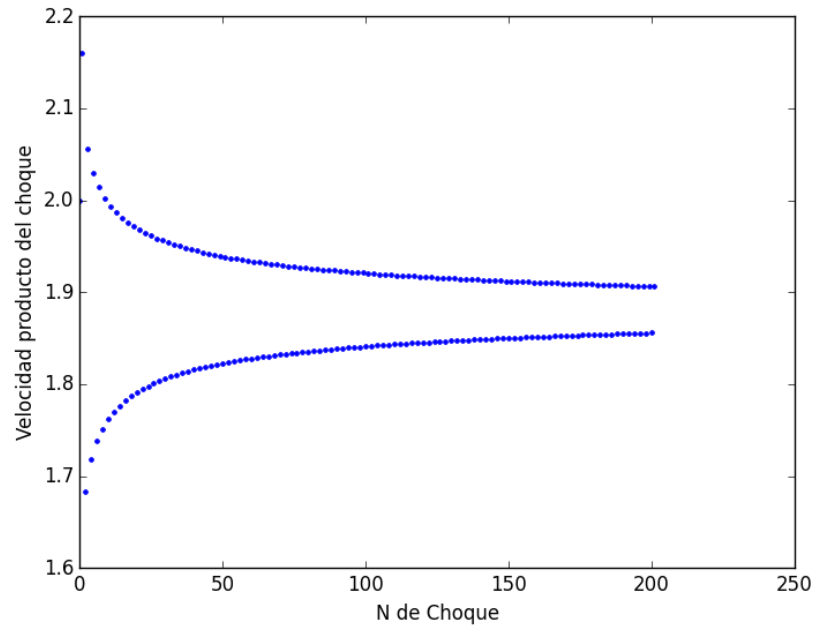


Figura 2: Relación entre la velocidad con la que rebota la partícula y el número del choque a $\omega = 1,67$

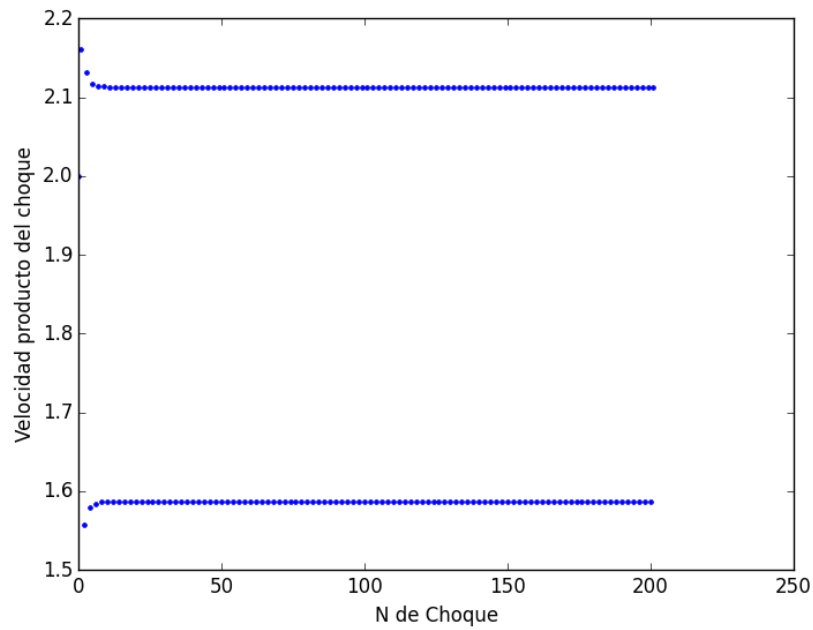


Figura 3: Relación entre la velocidad con la que rebota la partícula y el número del choque a $\omega = 1,69$

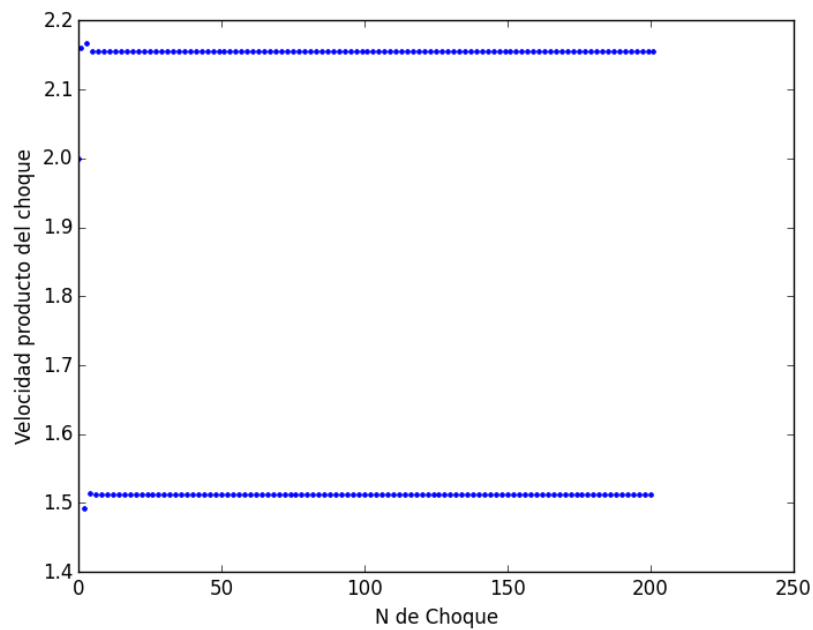


Figura 4: Relación entre la velocidad con la que rebota la partícula y el número del choque a $\omega = 1,7$

3.5. Velocidades luego del tiempo de relajación

Para esta parte se buscarán valores de velocidad luego de 2 veces el tiempo de relajación para distintos valores de ω entre 1.66 y 1.79. Se realizó una iteración de 20 entre valores

de ω desde 1.66 a 1.79 con la misma separación y se guardaron las velocidades a partir del rebote 120 hasta el 170, dado que el $N_{Relax} = 60$. Con esto, las iteraciones eran bastante estables dentro de los propios estados en los que estaban. Se ve de la parte anterior que ocurre una bifurcación entre 1.66 y 1.7, pero más adelante ocurren varias bifurcaciones más como se observa en la figura 5.

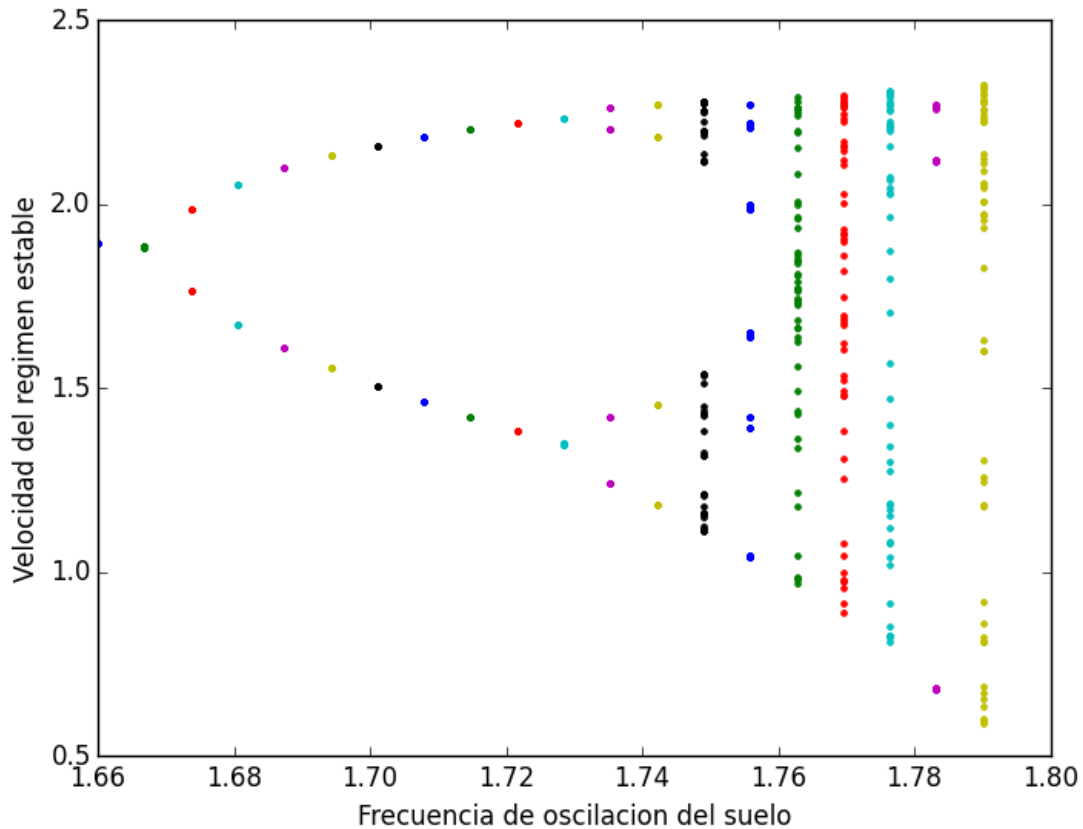


Figura 5: Relación entre velocidades en el régimen estable y la frecuencia de oscilación del suelo

El programa que se usó corresponde a "n3.pyz" realiza una iteración para cada valor de ω definidos anteriormente, donde se obtiene los valores de velocidad para la iteración 120 hasta la 170. También obtiene las posiciones, velocidades y el tiempo con para las partes anteriore dado que se ocupó la estructura del programa anterior. También obtiene el ω para cada iteración (que es el mismo) solo para poder graficarlo. Finalmente grafica todas las velocidades obtenidas con respecto a los ω para cada iteración, de manera que quedan todos graficados (con distintos colores en cada caso) en la misma figura.

Lo que se obtiene de este proceso corresponde a una estabilidad en la velocidad en un principio, lo cual puede significar que rebota en la misma parte del ciclo del movimiento del suelo. Luego la velocidad se estabiliza en dos puntos que podría ser que siempre choca en 2 puntos

fijos. Luego empiezan a aumentar los valores estables de la velocidad, lo que puede hacer pensar que son inestables. De cualquier manera, se observa que están más o menos focalizados en ciertas áreas, lo cual puede significar que existe cierta repetitividad en los valores y que no cambiaría mucho aunque se graficaran muchos más valores después de la estabilidad.

4. Conclusión

De este análisis podemos ver que cuando se agregan modelamientos de fenómenos a una situación física (en este caso, agregar disipación a choques de una partícula en una superficie sometida a un forzamiento) pueden ser modelados y se pueden observar fenómenos interesantes. También se puede observar que, en este caso, los fenómenos de forzaje y disipación se complementaron bastante, de manera que la partícula no disipaba toda su energía ni ganaba energía infinita por el forzaje. En particular, se nota que para ciertos valores de frecuencia de forzaje existe cierta periodicidad y que no es un sistema completamente caótico. Además, la situación inicial (la velocidad inicial de la partícula era el parámetro más relevante en este caso) tendía a no importar mucho para el resto del proceso (salvo que cumpliera las condiciones físicas de no ser menor a la velocidad del suelo cuando está pegada a este). Se puede destacar el rol de las herramientas computacionales en este tipo de problemas, ya que se obtienen resultados de iteraciones bastante rápido. Hacer el proceso manualmente para cada iteración no es algo sencillo. Siempre que se defina un buen cálculo dada una posición, velocidad y tiempo inicial, las iteraciones no deberían fallar y se pueden obtener resultados interesantes, como posibles resonancias de la pelota con respecto al suelo.