

**UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE FÍSICA**

ALGORITMO EFICIENTE PARA SIMULACIÓN DE MEDIOS GRANULARES DENSOS

TESIS PARA OPTAR AL GRADO DE MAGÍSTER
EN CIENCIAS, MENCIÓN FÍSICA

JUAN SEBASTIÁN GONZÁLEZ BRIONES

PROFESOR GUÍA
RODRIGO SOTO BERTRÁN

MIEMBROS DE LA COMISIÓN EVALUADORA
DINO RISSO ROCCO
ALEJANDRA ÁLVAREZ VALLEJOS
NICOLÁS MUJICA FERNÁNDEZ
GONZALO GUTIÉRREZ GALLARDO

SANTIAGO DE CHILE
DICIEMBRE 2008

Resumen

En este trabajo presentamos un nuevo algoritmo para simulación de medios granulares densos que incluye partículas estáticas en presencia de gravedad.

El modelo consiste en esferas duras inelásticas, para el cual contamos con un algoritmo de simulación eficiente. Extendemos la descripción de las partículas de manera de simular sólo las relevantes a la dinámica macroscópica del sistema.

Presentamos tres casos de estudio: deposiciones sobre un suelo estático, avalanchas en un plano rugoso inclinado y el flujo en un tambor rotante.

Estudiamos cómo dependen los observables físicos de los parámetros del algoritmo en tres casos: la fracción de empaquetamiento de esferas iguales, el ángulo de detención para avalanchas sobre un suelo rugoso y el perfil de velocidad en un tambor rotante.

Los resultados obtenidos son prometedores y muestran que el modelo de simulación conducida por eventos es útil para simular flujos densos, siempre y cuando se disipe energía de forma adecuada. La nueva extensión permite simular sistemas de interés en un tiempo computacional mucho menor que con métodos actuales.

Concluimos mostrando diferentes sistemas que pueden ser simulados con nuestro algoritmo y proponemos futuras líneas de investigación.

Agradecimientos

En primer lugar, al responsable de que hoy sea magister en física y no filósofo: Nelson Zamorano. Mi gusto por la física empezó en la Escuela de Verano hace como 10 años. Sin sus consejos, los dulces que me robe de su oficina y la confianza que me tuvo, todo habría sido infinitamente más difícil.

A la comisión por sus invaluable comentarios y el tiempo que me dedicaron a fin de mejorar esta tesis.

A Eric Clement por acogerme en Paris y salvarme de los parisinos, que como dice Bolaño, “son unos canibales”.

A Rodrigo que me soportó estos dos años y me mostro un camino cuando estaba perdido.

A Dino por todas las preguntas capciosas y las conversaciones interminables sobre el algoritmo.

A ella, que me hace cambiar de continente todos los años.

A mis padres por la fe ciega que tuvieron en que era buena idea que yo estudiara física y nunca dejaron de apoyarme.

Por último, quiero agradecer a FONDECYT y CIMAT por el apoyo financiero sin el cual este trabajo no podría haberse realizado.

Índice general

1. Introducción	1
1.1. Medios Granulares	1
1.2. Teoría	2
1.3. Simulaciones	3
1.3.1. Dinámica Molecular	3
1.3.2. Dinámica de Contactos	4
1.3.3. Dinámica molecular llevada por eventos	5
1.4. Objetivos de la Tesis	6
1.5. Estructura del Trabajo	6
2. Dinámica Molecular llevada por eventos	7
2.1. Modelo	7
2.2. Algoritmo	8
2.3. Resultados Anteriores	9
2.4. Problemas	10
2.4.1. Límite estático	11
2.4.2. Múltiples contactos	11
2.4.3. Colapso Inelástico	11
3. Algoritmo para simulación de fases densas	15
3.1. Motivación	15
3.1.1. Solución eficiente a la pelota saltarina	16
3.2. Idea general	17
3.2.1. Dormir	18
3.2.2. Despertar	19
3.3. Implementación	21
3.3.1. Cambio de Estados	21
3.3.2. Predicción de Eventos	23
3.3.3. Choques	27
3.4. Constantes introducidas	27
3.5. Áreas de Aplicación y Limitaciones	28
4. Deposición	30
4.1. Introducción	30
4.2. Configuración	31
4.3. Protocolo	31
4.4. Resultados	32

4.5. Proyecciones	36
5. Avalancha gruesa	37
5.1. Introducción	37
5.2. Configuración	38
5.3. Resultados	40
5.4. Proyecciones	43
6. Tambor Rotante	46
6.1. Introducción	46
6.2. Sistema	47
6.3. Protocolo	48
6.4. Resultados	49
6.5. Proyecciones	51
7. Divertimento	52
7.1. Spreading	52
7.2. Reloj de Arena	53
7.3. Proyecciones	55
8. Conclusiones	56
8.1. Algoritmo	56
8.2. Deposición	57
8.3. Avalancha Gruesa	57
8.4. Tambor Rotante	57
8.5. Divertimento	58
8.6. Trabajo futuro	58

Índice de figuras

1.1.	Medios granulares en sus tres fases posibles: en la izquierda, un sistema bidimensional vibrado donde coexisten las fase gaseosa por debajo y líquida arriba; en la derecha, la foto de una duna del desierto de Vietnam en que la arena forma un cuerpo con características de sólido. Fotos tomadas de [7] y [8]. . .	2
2.1.	División esquemática del espacio para una simulación 2D. La partícula azul pertenece a la celda de al medio, que tiene vecinas a todas las celdas amarillas. La partícula azul solo puede a) chocar con una de las partículas en celdas vecinas o b) salir de la celda que la contiene. Estos son todos los eventos que debemos predecir.	9
2.2.	Izquierda arriba: configuración experimental. Izquierda abajo: (a,b) Imágenes promediadas de las estructuras cristalinas encontradas, (c) Simulación ED que presenta un cluster de estructura cuadrada, (d) Procedimiento de coarse-graining, las líneas verticales definen los elementos típicos del fluido. Derecha: Comparación de resultados para el perfil de densidad (primera columna) y de momentum (segunda columna) para experimentos (a,b), simulaciones con ED (b,c) y teoría (d,e). Diagramas y fotografías tomados de [28]	10
2.3.	Diagrama de las fuerzas en una deposición 2D. En ancho de la línea indica el módulo de la fuerza de contacto y la dirección. Este tipo de estado no es posible de alcanzar con ED. Simulación realizada con el método de elementos discretos (DEM) [16]. Fotografía tomada de la referencia [17].	12
2.4.	Bosquejo de un colapso inelástico. Existen condiciones iniciales para las cuales la partícula de la derecha y de la izquierda chocan alternadamente con la del medio hasta que la energía es totalmente disipada en un número infinito de choques.	13
2.5.	Últimas 200 partículas que colisionaron en negro, para una disipación menor que la crítica hay enfriamiento homogéneo (derecha) mientras que para disipaciones mayores se produce clustering y colapso inelástico (izquierda). Fotografías tomadas de la referencia [27].	13
3.1.	Trayectorias para una pelota saltarina. Teóricamente, la partícula se detiene en t^* despues de infinitos choques (línea sólida). Si las colisiones se vuelven elásticas para pequeñas velocidades, en t_{elastic} la pelota queda dando infinitos botes (línea segmentada). Nuestro método duerme a la partícula si la velocidad es suficientemente pequeña y a partir de t_{sleep} la partícula queda en reposo (línea punteada).	16

3.2.	Esquema de las posiciones mecánicamente estable y no estables para dormir a las partículas en 2D. De izquierda a derecha: la primera partícula está en un punto estable, por lo que puede dormirse, la segunda está acelerando hacia abajo, por lo que no puede dormirse. La tercera está en una configuración estable gracias a la partícula normal y puede dormirse. La última partícula no puede dormirse puesto que no esta en chocando con partículas sólidas	19
3.3.	a) Preparación de una columna de granos por una lluvia aleatoria bajo la acción de la gravedad. b) La columna está caracterizada por su radio R_0 y altura inicial H_0 . c) Después del colapso, el depósito final está caracterizado por la distancia a la que llega el flujo R_∞ y su altura H_∞ . Figura tomada de [19].	20
3.4.	Esquema de las dos posibles situaciones al predecir el tiempo de choque entre partículas normales, la flecha representa la trayectoria de la partícula izquierda en el sistema de referencia de la partícula derecha. En el caso superior, las partículas se intersectan dos veces (estrellas rojas) y en el caso inferior no hay intersección.	24
3.5.	Diagrama de los tres posibles casos al predecir los tiempos entre partículas con distinta gravedad efectiva: la ecuación puede tener 4, 2 ó 0 soluciones reales, que corresponden a las tres formas posibles de intersectar una parábola con un círculo.	25
3.6.	Distancia entre los centros de masa para una partícula sólida y una normal como función del tiempo, que corresponde al primer caso de la figura 3.5. El polinomio entrega 4 soluciones que son los instantes en que entra y sale de la otra partícula. Las intersecciones de la línea sólida con la punteada indican las cuatro soluciones a la ecuación 3.3	26
3.7.	Diferentes configuraciones experimentales de flujos granulares densos. Nuestro algoritmo es de utilidad en b), c), e) y f) donde los perfiles de velocidad presentan una cola exponencial. Diagrama tomado de la referencia [35]. . . .	29
4.1.	Foto para los tiempos $t = 0\sqrt{d/g}; 50\sqrt{d/g}; 100\sqrt{d/g}$ (ordenados de izquierda a derecha) de una de nuestras deposiciones. Las partículas moradas pertenecen al suelo estático. Los distintos tonos de azul dependen de la energía cinética de las partículas mientras que el azul oscuro indica las dormidas.	32
4.2.	Fracción de empaquetamiento como función de las cotas para dormir y despertar. El valor del observable físico varía en un 1 % mientras nuestras cotas varían un orden de magnitud.	34
4.3.	Costo computacional para deposiciones. En el eje horizontal las cotas para dormir y despertar y en el eje vertical el número de eventos totales de la simulación.	34
4.4.	En el eje horizontal el tiempo y en el eje vertical el número de partículas en cada estado. Se aprecia que el número de partículas normales permanece acotado.	35
4.5.	En el eje horizontal el tiempo de la simulación y en el eje vertical el tiempo de cálculo en minutos. Como se aprecia, si no se duermen las partículas, el tiempo aumenta cuadráticamente.	35
5.1.	El número de Froude $\langle v \rangle / \sqrt{gd}$ en función de altura escalada por h_{stop} para distintas configuraciones experimentales, tomado de [35]	38

5.2.	Izquierda: Configuración experimental para estudiar h_{stop} . El flujo se regula por la altura de salida para el contenedor [35]. Derecha: Configuración de nuestras simulaciones para estudiar θ_{stop} . Una vez fijado el número de partículas en el sistema, el flujo se regula por el ángulo que está dado por la dirección de la gravedad. Hay condiciones de borde periódicas en x	39
5.3.	Foto con el obturador abierto para una avalancha de 10000 partículas en una cama de 40×10 diámetros.	39
5.4.	Evolución de la energía cinética promedio $\langle E_k \rangle = \frac{1}{2N} \sum_i mv_i^2$ en función del ángulo para una avalancha de $N = 2000$ partículas mientras se disminuye el ángulo a distintos θ	41
5.5.	Evolución de la energía para una avalancha de 2000 partículas mientras se disminuye el ángulo a distintos θ , esta vez graficado en función del tiempo de la simulación.	42
5.6.	Curva $\theta_{\text{stop}}(h)$ para nuestro sistema. Encontramos tres regímenes distintos: el sistema no fluye bajo $\theta_{\text{stop}}(h)$, fluye a velocidad constante sobre $\theta_{\text{stop}}(h)$, y flujos acelerados para ángulos muy grandes. Los errores para la altura son menores que los símbolos, mientras que para determinar el ángulo sólo realizamos una simulación, por lo que es difícil obtener una estadística.	43
5.7.	Curva $h_{\text{stop}}(\theta)$ para diferentes configuraciones experimentales [35].	44
5.8.	Evolución de la energía para una avalancha de 2000 partículas mientras se disminuye el ángulo a una tasa constante para diferentes valores de $V = V_{\text{sleep}} = V_{\text{wakeup}}$	45
6.1.	Foto para un instante de nuestras simulaciones del tambor rotante con 2000 partículas una vez alcanzado el régimen estacionario. Los colores representan la velocidad de la partícula de azul a rojo en el sistema de referencia rotado, las partículas dormidas están en azul oscuro, mientras que las que no pueden despertar y forman el borde en morado. El tambor gira en sentido horario.	47
6.2.	Configuración experimental y sistema de coordenadas usado experimentalmente. Imagen tomada de [35].	48
6.3.	Perfil de velocidades experimentalmente obtenidas en [35]. $D/d = 150$, $w/d = 4,66$; del fondo hacia arriba $Q = 5,25; 10,5; 15,75; 21; 30,75; 41,25$. En el inset, perfiles de velocidad para simulación de dinámica de contactos 2D [35]: $D/d = 150$ y $Q = 21; 30,75; 51,75$	49
6.4.	Perfil de velocidad para distintas v_{wakeup}	50
7.1.	Foto para tres instantes de un spreading. Arriba se ve el primer desprendimiento de los bordes; al medio se aprecia la avalancha que se forma; abajo, la configuración final a la que llega el sistema, con ambos ángulos de reposo iguales. El color representa la energía cinética, entre más oscuro, mayor energía	53
7.2.	Una de las configuraciones iniciales que se estudiaron con nuestro método y con dinámica de contactos.	53
7.3.	Probabilidad para la diferencia de las posiciones finales con ambos métodos.	54
7.4.	Fotos para una sección del reloj de arena en nuestras simulaciones, de izquierda a derecha avanza el tiempo. Las partículas están colorizadas según su altura inicial.	54

7.5. Resultados experimentales para flujos a traves de un orificio. Foto tomada de [37]	55
---	----

Capítulo 1

Introducción

En este capítulo describimos los medios granulares, presentamos un breve resumen de los métodos teóricos que se usan para su estudio y que tipo de simulaciones se realizan. Finalmente presentamos los objetivos de la tesis y la estructura del trabajo.

1.1. Medios Granulares

Los medios granulares están presentes en cada aspecto de nuestras vidas: desde el cemento que construye las ciudades, hasta las nueces que acompañan nuestros cócteles.

Una definición más precisa y menos poética, sería la de un conglomerado de partículas sólidas macroscópicas (cuya energía típica es $E \gg k_B T$) caracterizadas por interacciones de choque y una pérdida de energía cada vez que interactúan. Dependiendo de la energía promedio de las partículas, los medios granulares pueden presentar comportamientos propios de las distintas fases de la materia: gas, líquido y sólido. Incluso se pueden presentar juntas en un mismo sistema (ver figura 1.1). Algunas de sus características se presentan en fenómenos como la segregación por tamaño [1], formación de patrones al ser sometidos a vibraciones [2], formación de rollos convectivos [3] y ondas de superficie [4], entre otros.

Los medios granulares son importantes comercialmente en aplicaciones farmacéuticas, mineras y de la industria alimenticia. Es por esto que desarrollar modelos que permitan predecir su comportamiento es de una gran importancia tanto práctica como teórica.

En el presente trabajo consideraremos medios granulares secos, donde el fluido que los envuelve (por ejemplo aire) tiene un efecto despreciable sobre la dinámica de los granos.

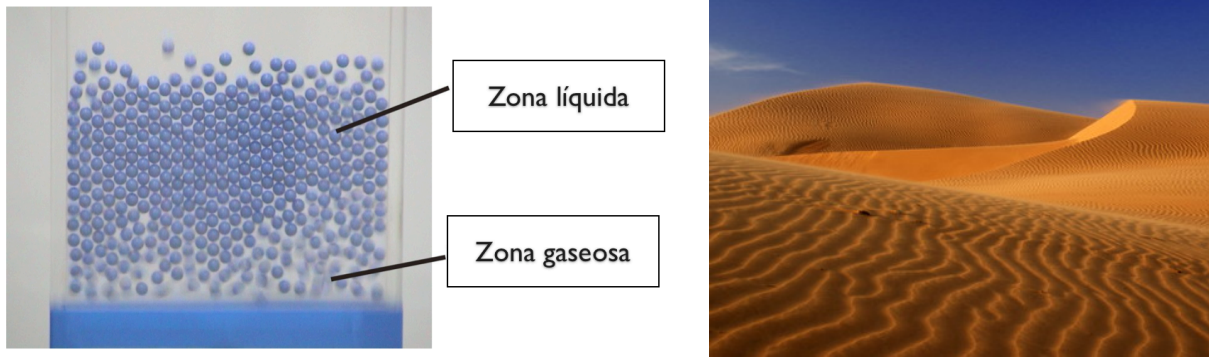


Figura 1.1: Medios granulares en sus tres fases posibles: en la izquierda, un sistema bidimensional vibrado donde coexisten las fase gaseosa por debajo y líquida arriba; en la derecha, la foto de una duna del desierto de Vietnam en que la arena forma un cuerpo con características de sólido. Fotos tomadas de [7] y [8].

1.2. Teoría

Hay dos aspectos importante que contribuyen a las propiedades únicas de los medios granulares y que hacen difícil su estudio: la temperatura no tienen ningún rol y las interacciones entre los granos son disipativas debido a la fricción e inelasticidad de las colisiones.

Uno de los roles importantes de la temperatura en sistemas clásicos es permitir explorar el espacio de fase. En un medio granular, $k_B T \sim 0$ impide esta exploración. Luego, a menos que existan perturbaciones externas, cada configuración metaestable dura indefinidamente, y no hay promedios sobre configuraciones cercanas en el espacio de fase.

La teoría cinética de gases granulares (baja densidad y disipación) está bien establecida [9]. Partiendo de la ecuación de Boltzmann y agregando disipación, se obtienen ecuaciones tipo hidrodinámicas para gases. Sin embargo, la analogía con los gases no llega muy lejos. Debido a que los choques entre las partículas presentan disipación, el único estado de equilibrio para un sistema granular al cual no se le inyecta energía es el reposo.

En sistemas densos, donde los contactos son permanentes y las correlaciones son de largo alcance, la descripción es compleja. En [10], se presenta el caso de los flujos densos en planos inclinados, para los cuales, en el caso de que la fricción sea pequeña, la teoría concuerda relativamente bien con los resultados numéricos y experimentales. Sin embargo, en el caso de flujos con fricción considerable, todavía no hay un marco teórico que explique los fenómenos observados.

Pese a esta falta de un marco teórico general, se han encontrado leyes fenomenológicas

para los sistemas granulares. Un ejemplo es la ley constitutiva para flujos densos propuesta por Jop et al. [5] que relaciona el roce efectivo del sistema con la tasa de cizalle.

1.3. Simulaciones

Las simulaciones computacionales de medios granulares son necesarias principalmente por tres motivos: no contamos con una teoría que prediga su comportamiento general, sino más bien muchas teorías que dependen del régimen en que se encuentre el medio; hacer experimentos puede resultar caro, lento o peligroso; obtener las trayectorias de las partículas en experimentos tridimensionales es extremadamente complejo, mientras que en las simulaciones se tiene acceso a todas las variables del sistema para cada instante de tiempo, permitiendo una descripción mucho más precisa.

A continuación describimos los tres métodos más usados en la literatura actual para la simulación de medios granulares. Por simplicidad nos remitiremos a los casos de esferas monodispersas o discos polidispersos.

1.3.1. Dinámica Molecular

Consiste en la integración numérica de las ecuaciones de Newton para un conjunto de partículas, en general, átomos o moléculas. Estas partículas interactúan por potenciales continuos. Una referencia ya clásica para dinámica molecular es el libro de Frenkel y Smit [6].

En el caso de esferas duras, estos potenciales sólo son efectivos cuando las partículas están en contacto. Dado que modelar la deformación de las partículas de forma real es extremadamente complejo, se supone que la cantidad importante que define la interacción entre dos partículas, que llamaremos 1 y 2, es la penetración $\delta = \frac{1}{2}(d_1 + d_2) - (\vec{r}_1 - \vec{r}_2) \cdot \vec{n}$, donde \vec{r}_i es la posición de la partícula i y d_i su diámetro.

A partir de δ se calculan las fuerzas a las cuales están sometidas las partículas que están traslapadas. Estas fuerzas deben dar cuenta de la inelasticidad de los choques, del volumen excluido de las partículas y de la fricción entre ellas. Una vez definido el modelo a usar para las tres fuerzas, se integran las ecuaciones de Newton a intervalos de tiempo dt fijos. El modelo para esferas elásticas fue derivado por Hertz [11], donde la fuerza de restitución depende de la deformación y de las propiedades mecánicas del material, caracterizadas por el módulo de Young Y y el coeficiente de Poisson ν :

$$F = \frac{2Y\sqrt{R^{\text{eff}}}}{3(1-\nu^2)}\delta^{3/2}, \quad (1.1)$$

donde $R^{\text{eff}} = (d_i d_j)/2(d_i + d_j)$ es el radio efectivo de las partículas que chocan.

Este resultado fue posteriormente generalizado para partículas viscoelásticas [14]:

$$F = \frac{2Y\sqrt{R^{\text{eff}}}}{3(1-\nu^2)}(\delta^{3/2} + A\sqrt{\delta}\dot{\delta}), \quad (1.2)$$

con la constante A siendo una función de la viscosidad del material.

Para que la simulación sea estable, es necesario que el intervalo dt sea mucho menor que la duración del tiempo típico de choque entre dos partículas, en general, un décimo de $\tilde{\Delta}t$.

La ventaja de este método es su simplicidad: una vez definidas las reglas para las fuerzas, hay que calcular el traslape entre todas las partículas e integrar.

Sin embargo, debido a la restricción sobre el tiempo de integración, se modelan partículas artificialmente blandas, ya que al durar más el tiempo de colisión, no es necesario un dt tan pequeño.

Otro de los problemas de dinámica molecular es la complejidad de simulación del roce estático. Dado que éste sólo tiene una cota superior, para calcular esta fuerza es necesario resolver las ecuaciones de Newton para cada partícula. El costo computacional para cada paso dt es de $\mathcal{O}N$, donde N es el número de partículas en la simulación.

1.3.2. Dinámica de Contactos

El método de la dinámica de contactos fue desarrollado por Moreau y Jean [12] durante los años 80, con el fin de enfrentar el problema de las irregularidades matemáticas encontradas al modelar sistemas mecánicos, en los cuales se toma en cuenta la restricción de impenetrabilidad de las partículas. Esta restricción genera saltos en el valor de la velocidad, cada vez que ocurren colisiones entre partículas, lo que se traduce en discontinuidades en las aceleraciones. A diferencia de otros métodos, como de dinámica molecular, que resuelven este problema regularizando matemáticamente el sistema a través de la definición de una fuerza de interacción, la dinámica de contactos trata el sistema a nivel de las velocidades, evitando de esta forma, la introducción de una ley de interacción. Este método consiste en determinar, las velocidades y reacciones después de una colisión, a partir de las velocidades y reacciones conocidas, antes de dicha colisión, de manera tal de satisfacer en forma simultánea,

la conservación del momentum total y las leyes de contacto, las que tienen relación con la impenetrabilidad de las partículas y la fricción mutua entre ellas [13].

El algoritmo consiste en que para cada paso temporal dt se calculan todos los contactos (i.e. traslapes) y se escribe una ecuación para las fuerzas imponiendo que al final del paso de integración las partículas no estén traslapadas. Para resolver estas ecuaciones, que no tienen solución analítica, se usa un método iterativo. De esta forma es posible encontrar las fuerzas de restitución junto con el valor del roce estático que debe cumplir la condición de Coulomb $F_r \leq \mu N$ [18].

Este método es el que se utiliza mayoritariamente en el estudio de flujos granulares densos donde la fricción tiene un rol preponderante [19].

La ventaja de este método es que el dt es mucho mayor que en dinámica molecular, disminuyendo el costo computacional. La desventaja es que el costo computacional escala con el número de contactos del sistema. De esta manera, realizar simulaciones en 3 dimensiones y con un número considerable de partículas se torna computacionalmente intratable. Al igual que el método anterior, el costo computacional para cada paso dt es de $\mathcal{O}N$.

1.3.3. Dinámica molecular llevada por eventos

La dinámica molecular llevada por eventos, Event Driven (ED) en inglés, consiste en integrar la trayectoria del sistema en pasos dados por la dinámica de éste y no a pasos temporales fijos, disminuyendo drásticamente el costo computacional.

En el caso de esferas duras, donde los potenciales son discontinuos, no es necesario integrar el sistema a intervalos de tiempo fijo. Debido a que las partículas entre los choques se mueven balísticamente, es posible avanzar el sistema de choque en choque, o mejor dicho, de evento en evento [20].

Este método es mucho más eficiente que los dos anteriores para bajas densidades. Sin embargo, cuando la densidad aumenta y con ello el número de choques en el sistema, el método se vuelve altamente ineficiente. Este es el problema que avordaremos en este trabajo.

En el próximo capítulo explicaremos en detalle este algoritmo.

1.4. Objetivos de la Tesis

Los objetivos de esta tesis son: presentar una extensión al modelo ED [20] para simular partículas estáticas en presencia de gravedad, aplicar esta extensión a tres casos de estudio validando el modelo y estudiar la eficiencia computacional de la extensión.

La idea detrás de implementar estas partículas es lograr simular eficientemente flujos granulares densos donde coexisten regiones sólidas con capas fluidizadas, por ejemplo, en el caso del tambor rotante.

1.5. Estructura del Trabajo

En capítulo 2 presentamos el algoritmo Event Driven original. En el capítulo 3 presentamos la extensión con todos los detalles de la implementación a fin de que pueda ser reproducida por terceros. En los capítulos 4, 5 y 6 presentamos resultados de los 3 casos estudiados. En el capítulo 7 presentamos *divertimentos* que hemos hecho con nuestro algoritmo y que podrían servir como base para posteriores investigaciones. Finalmente en el capítulo 8 se presentan las conclusiones.

Capítulo 2

Dinámica Molecular llevada por eventos

En este capítulo presentamos el algoritmo original de ED en detalle. Una vez explicado el algoritmo, mostramos resultados ejemplares obtenidos con él para luego señalar los problemas que presenta y que limitan su rango de aplicación.

2.1. Modelo

El modelo de esferas duras inelásticas considera partículas que interactúan a través de choques binarios instantáneos, ya sea entre ellas o con paredes. Cuando dos partículas sin fricción chocan, las velocidades postcolisionales están determinadas por las reglas de choque, que en el caso de masas y radios iguales son:

$$\vec{v}_i' = \vec{v}_i \pm \frac{1 + \alpha}{2} [(\vec{v}_2 - \vec{v}_1) \cdot \hat{n}] \hat{n} \quad (2.1)$$

donde α es el coeficiente de restitución, \hat{n} es el vector unitario que apunta del centro de la partícula 1 a la 2 y \vec{v}_i es la velocidad precolisional de la partícula i . De la ecuación anterior se puede despejar α , que resulta ser la razón entre las componentes normales de la velocidad relativa antes y después del choque

$$\alpha = - \frac{(\vec{v}_2' - \vec{v}_1') \cdot \hat{n}}{(\vec{v}_2 - \vec{v}_1) \cdot \hat{n}} \quad (2.2)$$

Los choques con las paredes dependen del tipo de pared que se esté simulando, sean estas móviles o fijas, termales o disipativas. El caso de una pared disipativa se logra escribiendo la regla de choque con una partícula de masa y radio infinitos.

2.2. Algoritmo

El método de ED consiste en predecir, dado el estado de un sistema, el tiempo de la próxima evento o colisión, avanzar el sistema analíticamente hasta ese instante, calcular las velocidades postcolisionales y repetir. Dado que no hay una discretización temporal fija, el método es más preciso que dinámica molecular. Con ciertas optimizaciones [22] es posible reducir el costo computacional de atender cada evento a $\mathcal{O}(\log N)$.

La versión más básica del algoritmo avanza todo el sistema cuando se produce un choque, haciendo que el costo computacional sea $\mathcal{O}(N)$. Dado que los choques son binarios e instantáneos a lo más hay sólo dos partículas en contacto en cada evento, por esto, no es necesario evolucionar todo el sistema a un estado en que solo 2 partículas chocan. La solución consiste en agregarle a la descripción de las partículas el último tiempo de choque. De esta forma, para evolucionar el sistema se avanzan solo las partículas involucradas en el choque puesto que todas las demás continúan con la misma dinámica que tenían. Esto hace que la simulación sea asíncrona, y por lo tanto más compleja para medir observables, pero mucho más eficiente.

De esta manera, cada partícula está definida por 3 variables, dos vectoriales y una escalar: Posición \vec{r} , velocidad \vec{v} y tiempo en que la partícula cambió de estado por última vez t_{ant} . Los valores de \vec{r} y \vec{v} son los que tenía la partícula en t_{ant} . Dadas las posiciones y velocidades en el tiempo t_{ant} es posible calcular la posición y velocidad actual para cualquier tiempo $t > t_{ant}$ con las ecuaciones cinemáticas con o sin gravedad.

Cuando se atiende el choque entre dos partículas hay que modificar las posiciones a las posiciones actuales y actualizar las velocidades según la regla de choque. Una vez hecho eso se deben calcular todos los posibles choques de las partículas involucradas con el resto del sistema. Como es poco probable que partículas lejanas choquen, se desarrolla un esquema de celdas de manera que la predicción sólo se realiza con las partículas de celdas vecinas como se ve en la figura 2.1. Debido a esta modificación, es necesario agregar a la simulación los eventos de cambio de celda, eventos en que no se modifica el estado de la partícula pero sí la estructura que las mantiene ordenadas en el espacio.

Una vez calculados los tiempos de todos los próximos eventos de las partículas involucradas, se agregan en una cola de prioridad, que permite encontrar en orden $\mathcal{O}(\log N)$ cual es el tiempo mínimo de los próximos eventos en la simulación.

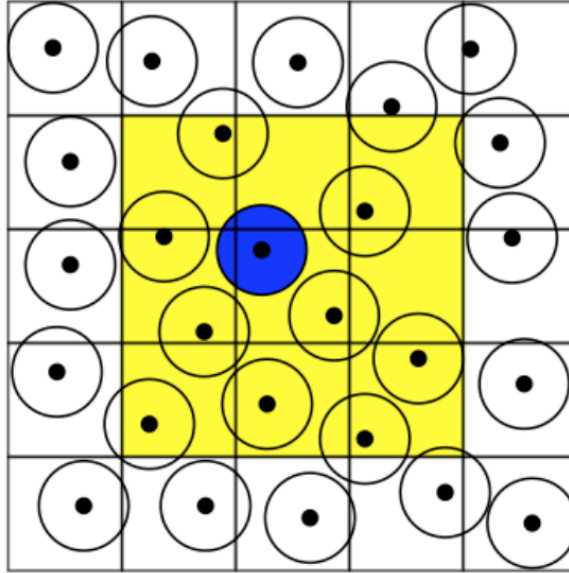


Figura 2.1: División esquemática del espacio para una simulación 2D. La partícula azul pertenece a la celda de al medio, que tiene vecinas a todas las celdas amarillas. La partícula azul solo puede a) chocar con una de las partículas en celdas vecinas o b) salir de la celda que la contiene. Estos son todos los eventos que debemos predecir.

Detalles de estas optimizaciones se pueden encontrar en las referencias [23] y [22].

2.3. Resultados Anteriores

El algoritmo ED ha sido usado en una gran cantidad de simulaciones, por ejemplo, las de Cordero y Soto [24] en el problema de la nuez de Brasil, Luding [26] en dinámica de cluster granulares y enfriamiento, McNamara [27] en gases granulares y colapso inelástico, Riso en transiciones de fase en medios granulares fluidizados [28] y Pöschel en gases granulares [29].

Debido a sus limitaciones, de las que hablaremos en la sección 2.4, los sistemas estudiados con ED han sido principalmente medios vibrados, donde las paredes le inyectan energía al sistema cada vez que las partículas chocan contra ellas, y gases granulares (en ausencia de gravedad), dejando de lado toda el área de flujos densos o sistemas estáticos.

Un resultado ejemplar es el de Clerc et al. [28], donde se estudia la transición de fase de un medio cuasi-unidimensional vibrado, en forma experimental, teórica y con simulaciones ED, mostrando como estos tres enfoques pueden complementarse para explicar fenómenos de la naturaleza.

El sistema estudiado se ve en la figura 2.2 y es un sistema granular vibrado desde abajo que presenta separación de fases.

Este sistema es ideal para ser simulado con ED, puesto que las partículas reciben energía por medio de la vibración del suelo y la disipan en los choques entre ellas y con las paredes, encontrándose siempre fuera del equilibrio.

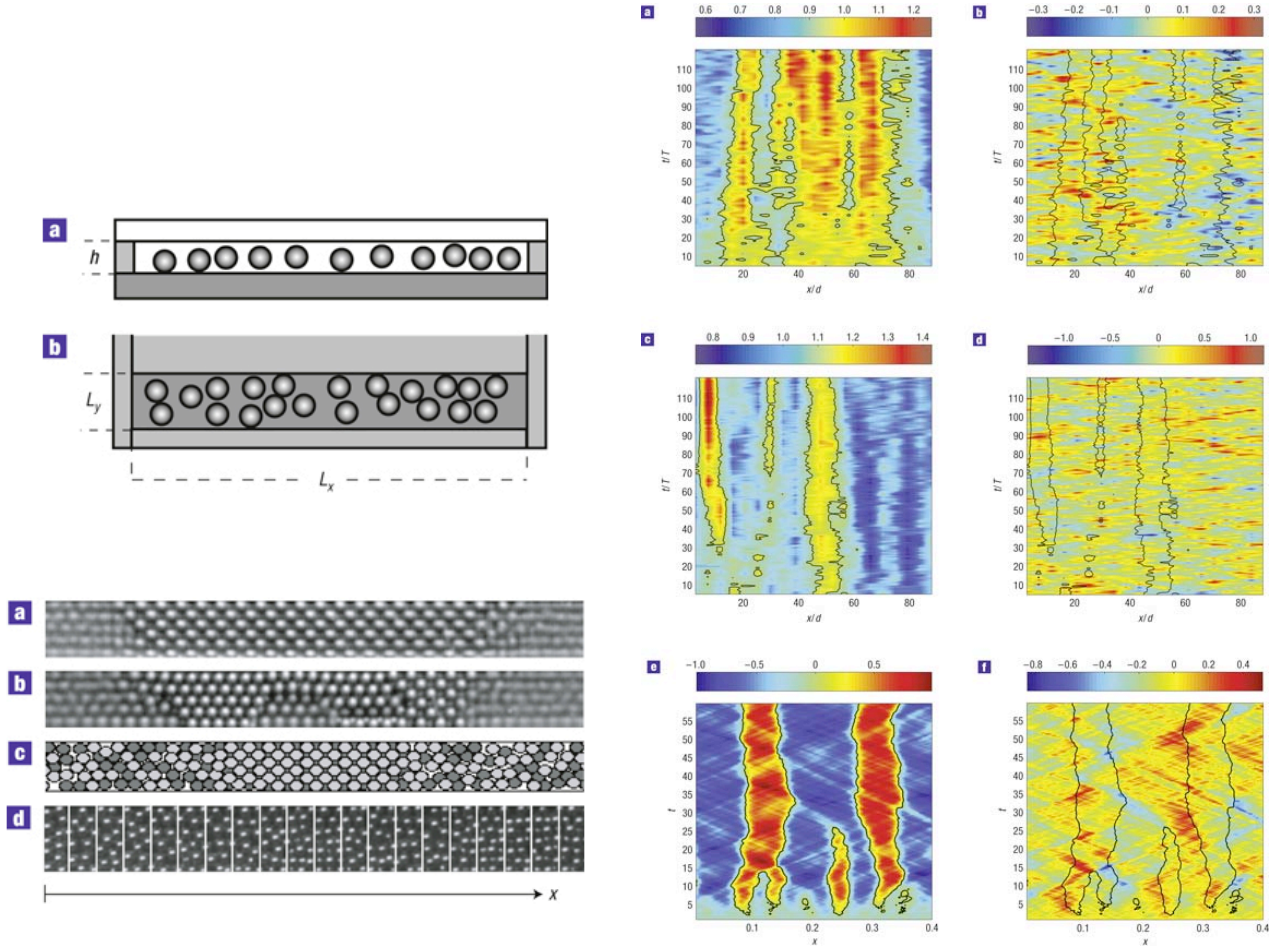


Figura 2.2: Izquierda arriba: configuración experimental. Izquierda abajo: (a,b) Imágenes promediadas de las estructuras cristalinas encontradas, (c) Simulación ED que presenta un cluster de estructura cuadrada, (d) Procedimiento de coarse-graining, las líneas verticales definen los elementos típicos del fluido. Derecha: Comparación de resultados para el perfil de densidad (primera columna) y de momentum (segunda columna) para experimentos (a,b), simulaciones con ED (b,c) y teoría (d,e). Diagramas y fotografías tomados de [28]

2.4. Problemas

Debido a que el modelo ED está pensado para simular sistemas gaseosos de baja densidad, donde los tiempos de contactos son pequeños comparados con la dinámica del sistema y los

choques mayormente binarios, su uso en sistemas que no cumplan lo anterior presenta 3 problemas graves [31]. Estos son: la ausencia de un límite estático; la imposibilidad para modelar múltiples contactos y el colapso inelástico. De algún modo todos los problemas tienen la misma causa: suponer que los choques tienen una duración nula.

2.4.1. Límite estático

La forma de modelar las partículas en ED es considerando que ellas están siempre en movimiento entre choques, y si corresponde, sujetas a la acción de la gravedad.

Debido a esto, no es posible que una partícula tenga velocidad nula por un tiempo arbitrario, haciendo imposible simular, por ejemplo, una partícula en reposo contra un suelo. De esta forma, simular pilas o en general configuraciones estáticas de medios granulares es imposible: la energía cinética en estos sistemas es siempre distinta de cero.

De esta manera, configuraciones como la de la figura 2.3 no se pueden simular con ED.

2.4.2. Múltiples contactos

Al considerar que el tiempo de la colisión entre las partículas es nulo, no pueden producirse choques entre más de dos partículas en el mismo instante. Esto que parece razonable en gases ideales o gases granulares, al ser llevado a flujos densos por ejemplo, deja de tener validez: en los flujos granulares densos las partículas experimentan contactos persistentes en el tiempo y se deslizan unas sobre otras.

Debido a esto, tampoco es posible realizar estudios de las cadenas de fuerza en medios granulares (como el de la figura 2.3), ya que en cada instante de tiempo, a lo más hay solo dos partículas en contacto.

2.4.3. Colapso Inelástico

Si se supone un coeficiente de restitución que no depende de la velocidad relativa de las partículas $\alpha = \text{const}$, puede ocurrir un efecto artificial llamado “*colapso inelástico*”: Considérese tres partículas idénticas que se mueven en una línea. Si el coeficiente de restitución es suficientemente pequeño, existen velocidades iniciales v_1 , v_2 y v_3 para las cuales las partículas sufren infinitas colisiones en un tiempo físico finito (Fig. 2.4) [27].

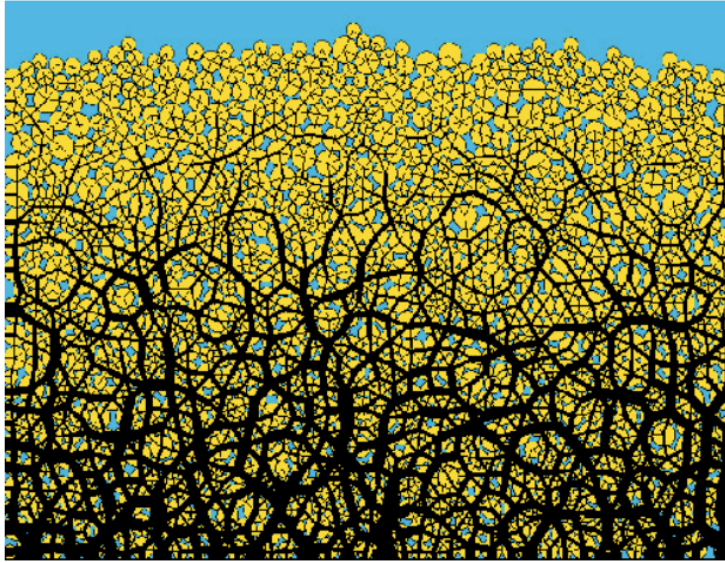


Figura 2.3: Diagrama de las fuerzas en una deposición 2D. En ancho de la línea indica el módulo de la fuerza de contacto y la dirección. Este tipo de estado no es posible de alcanzar con ED. Simulación realizada con el método de elementos discretos (DEM) [16]. Fotografía tomada de la referencia [17].

Este artefacto numérico se encontró al estudiar el enfriamiento de sistemas bidimensionales sin gravedad como el de la figura izquierda de 2.5. En esas simulaciones se deja evolucionar un sistema granular diluido y homogéneo sin gravedad. Dependiendo del coeficiente de restitución de las partículas se observan dos comportamientos cualitativamente diferentes: con poca disipación el sistema se enfría homogéneamente, mientras que sobre un valor crítico para la disipación, el sistema pierde la homogeneidad y desarrolla grumos o “cluster” de partículas con densidad alta y zonas de poca densidad. En las zonas de alta densidad, las partículas disipan la energía más rápido debido a que hay más choques de la figura 2.5. De esta forma, se producen líneas de partículas que se mantienen chocando cada vez a una velocidad menor haciendo que el número de eventos que la simulación debe atender diverja.

Es fácil ver que el colapso inelástico es algo inevitable en cualquier simulación ED que considere α constante. Cabe recalcar aquí que mientras que la formación de “clusters” de partículas en gases que se enfrían es un fenómeno físico, el colapso inelástico es un artefacto del tipo de simulación que se realiza, donde las colisiones se consideran instantáneas.

Suponer el coeficiente de restitución constante no concuerda con la mecánica del material. Un modelo más realista considera que a medida que la velocidad relativa v_g , del choque tiende a cero, el coeficiente de restitución tiende a uno $\epsilon_{v_g \rightarrow 0} = 1$. Es interesante notar que no todos estos modelos escapan del colapso inelástico, por ejemplo, el presentado por Ramírez et al.

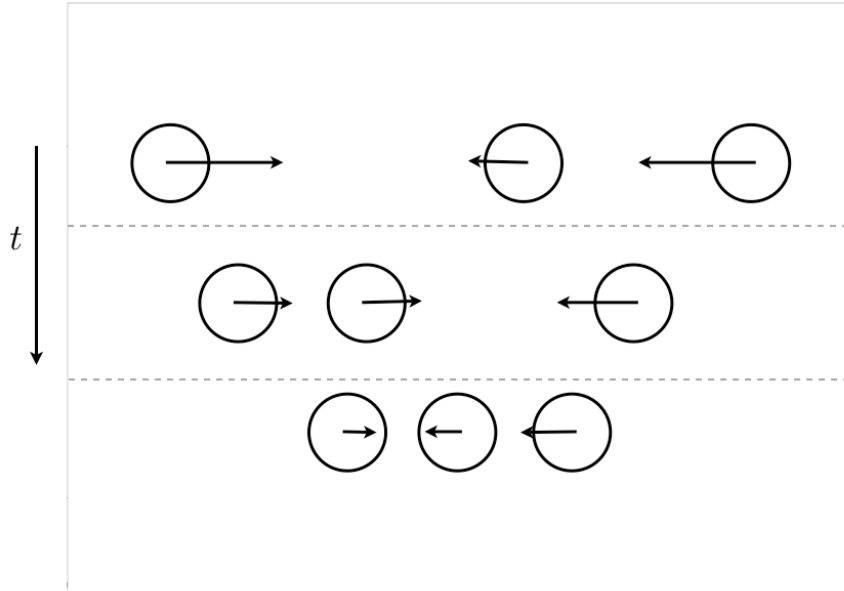


Figura 2.4: Bosquejo de un colapso inelástico. Existen condiciones iniciales para las cuales la partícula de la derecha y de la izquierda chocan alternadamente con la del medio hasta que la energía es totalmente disipada en un número infinito de choques.

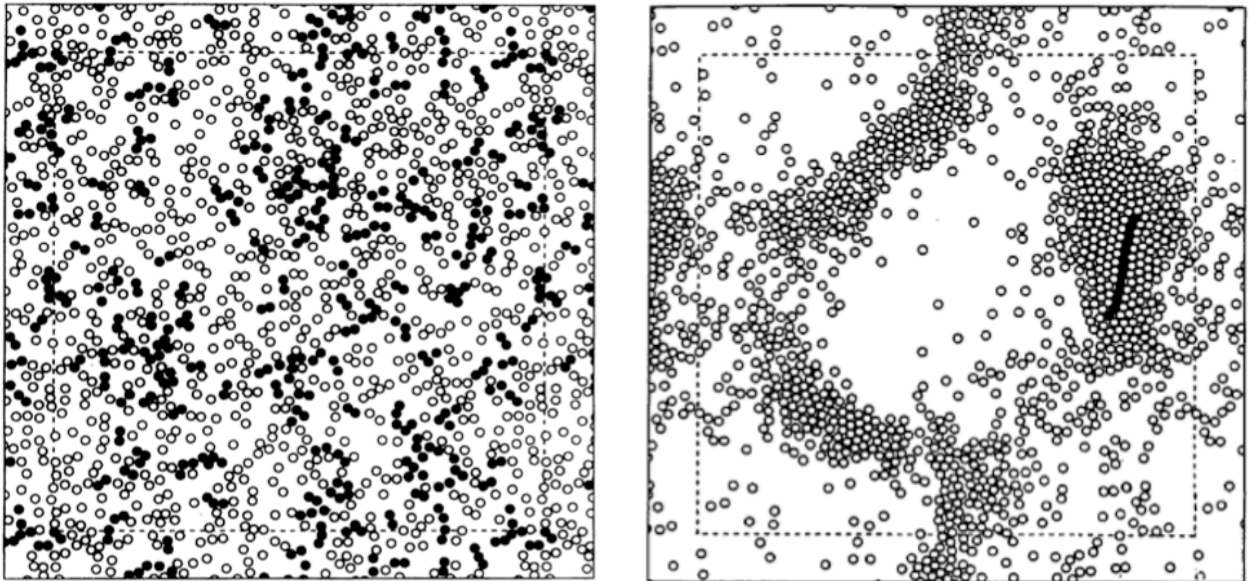


Figura 2.5: Últimas 200 partículas que colisionaron en negro, para una disipación menor que la crítica hay enfriamiento homogéneo (derecha) mientras que para disipaciones mayores se produce clustering y colapso inelástico (izquierda). Fotografías tomadas de la referencia [27].

[30] propone $\epsilon = 1 - \left(\frac{v_g}{v_{g0}}\right)^{1/5}$ pero también da lugar a colapso inelástico.

Una de las soluciones a este problema consisten en hacer elástico el choque entre las partículas una vez que su velocidad relativa sea suficientemente pequeña. Si bien se evita que

el número de eventos se vaya a infinito, esto hace que simular sistemas densos sea altamente ineficiente con ED ya que las partículas permanecen rebotando con una velocidad relativa pequeña que no afecta la dinámica general del sistema. Son estos choques los que consumen la mayor parte del tiempo computacional. Los otros dos problemas no han podido ser resueltos hasta el momento.

En el siguiente capítulo presentaremos una extensión al modelo ED que permite simular partículas estáticas. Al agregar una dinámica en que las partículas están detenidas, por un lado damos cuenta de que físicamente están quietas desde un punto de vista macroscópico, y por otro, no simulamos eventos en escalas temporales irrelevantes para la evolución del sistema que, en caso de hacerlo, como lo hacen los algoritmos actuales, consumirían la mayor parte del tiempo computacional.

Capítulo 3

Algoritmo para simulación de fases densas

En este capítulo presentamos la extensión que proponemos realizar al modelo ED para que este soporte partículas estáticas en presencia de gravedad. A partir de un modelo de juguete motivamos la necesidad y la ventaja de nuestro enfoque para luego detallar la implementación de esta idea.

3.1. Motivación

Como es sabido, una pelota que choca contra el suelo con velocidad v rebota con velocidad αv , donde α es el coeficiente de restitución. A pesar de que sabemos que el coeficiente de restitución depende de la velocidad, por consideraciones prácticas podemos tomarlo como una constante para un material dado. Un cálculo elemental muestra que si la altura inicial es h_0 , la próxima altura máxima será $\alpha^2 h_0$ y la n -ésima $\alpha^{2n} h_0$. Denotemos por τ_n el tiempo que pasa entre h_n y h_{n+1} . Es fácil mostrar que $\tau_n = \tau_0 \alpha^n$ y dado que la suma de τ_n es finita, ya que $0 < \alpha < 1$, se sigue que la partícula se detiene al cabo de un tiempo finito pero donde ha tenido infinitas colisiones.

Es por esto, que las simulaciones ED, que siguen la dinámica de la partícula de choque en choque, no pueden simular más allá que un cierto tiempo t^* . Las extensiones actuales a ED mantienen a la partícula dando infinitos botes con una velocidad pequeña, para evitar el colapso inelástico, y con un coeficiente de restitución $\alpha = 1$, como se ve en la figura 3.1. Esto hace que las simulaciones sean altamente ineficientes, ya que continuamente atienden una dinámica que no afecta al sistema desde un punto de vista macroscópico (la pelota da botes en el *mismo* lugar) y más aún, dejan un remanente de energía que podría afectar el comportamiento global del sistema. Por ejemplo, en el caso de que el suelo tuviera un pequeño ángulo, la partícula comenzaría a deslizar cuando físicamente debería estar detenida.

3.1.1. Solución eficiente a la pelota saltarina

Lo que nosotros proponemos es detener artificialmente la partícula cuando su velocidad sea menor que una cierta cota. Esta sería ahora una partícula en estado “sólido”. Con esto logramos sobrellevar dos de los problemas asociados largamente ED: la energía del sistema es efectivamente nula y no caemos en el colapso inelástico puesto que la partícula es detenida antes de que este suceda (ver Fig. 3.1).

Notemos que al detener artificialmente a la partícula estamos cometiendo dos errores en la descripción. Por un lado, detenemos a la partícula *antes* de que se produzca el colapso inelástico. Por otro, dejamos a la partícula con una pequeña separación con el suelo por motivos que se explicarán más adelante. Con las condiciones adecuadas para cambiar de estado a las partículas se pueden controlar estos errores.

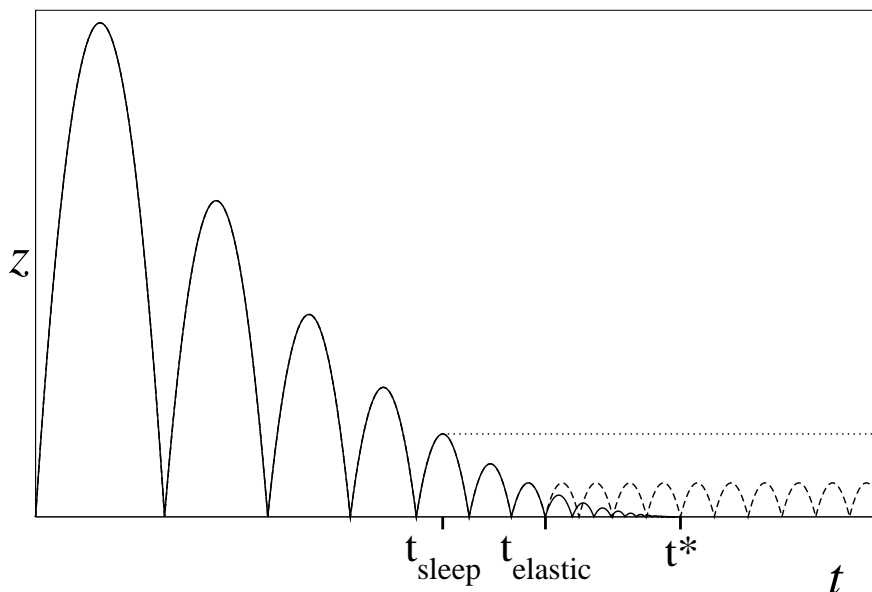


Figura 3.1: Trayectorias para una pelota saltarina. Teóricamente, la partícula se detiene en t^* después de infinitos choques (línea sólida). Si las colisiones se vuelven elásticas para pequeñas velocidades, en t_{elastic} la pelota queda dando infinitos botes (línea segmentada). Nuestro método duerme a la partícula si la velocidad es suficientemente pequeña y a partir de t_{sleep} la partícula queda en reposo (línea punteada).

Con este modelo de juguete hemos visto que tanto por razones de eficiencia computacional como para respetar la física de la simulación, debemos cambiar la dinámica de ciertas partículas. Llamaremos “dormir” a este cambio en la dinámica de la partícula puesto que las partículas en estado “sólido” dejan de participar activamente en los eventos del sistema. Al cambio de estado contrario, es decir, cuando la partícula recupera la dinámica “normal” lo

llamaremos “despertar”. De esta forma, lo que sigue del trabajo nos dedicaremos a explicar cómo, bajo qué condiciones y de qué manera vamos a dormir y despertar a las partículas en una simulación ED. Veamos a continuación la generalización de esta idea.

3.2. Idea general

Lo que buscamos es ser capaces de dormir y despertar a las partículas bajo ciertas condiciones para que por un lado no agreguen energía al sistema de forma irreal y por otro no estén dando botes que hacen ineficiente la simulación.

En términos más generales, necesitamos implementar un algoritmo ED con coexistencia de dinámicas. En este contexto la dinámica de una partícula es una etiqueta que determina el tipo movimiento que obedece y las reglas de choque con otras partículas.

Para permitir distintas dinámicas debemos hacer las siguientes modificaciones al actual método ED:

- explicar cómo realizar el cambio de estado de las partículas: puesto que nuestro sistema avanza por colisiones, el cambio de estado debe ser programado inteligentemente para que las partículas no queden en contacto, ya que esto haría que tuviéramos que resolver choques entre múltiples partículas, un problema que no está definido para las esferas duras.
- explicar la dinámica que siguen las partículas dormidas y encontrar las ecuaciones que hay que resolver para encontrar el tiempo de choque entre una partícula normal y una sólida.
- explicar cuáles van a ser las condiciones que se deben cumplir en la trayectoria de la partícula para decidir su cambio de estado.
- explicar cómo hacer los choques entre partículas de distintos estados. Debido a que las partículas dormidas forman conglomerados, modelamos el choque como si la partícula dormida tuviera masa infinita y disipara más energía que los choques normales.

Antes de continuar con la explicación de cómo hacer el cambio de dinámicas, debemos explicitar la clase de partículas con las cuales vamos a trabajar. Esto es porque las condiciones para dormir o despertar dependerán de la dimensión en que realicemos la simulación, la distribución de radios de las partículas y si tienen o no rotación.

Nuestro algoritmo partirá considerando partículas esféricas en 3 dimensiones, de igual radio, igual masa y sin rotación.

Para simplificar las comparaciones, usaremos unidades donde el módulo de la gravedad es $g = 1$, el diametro de las partículas es $d = 1$ y su masa $m = 1$. Esto define un tiempo característico $t = \sqrt{d/g} = 1$, una velocidad característica $\sqrt{gd} = 1$ y una energía típica $E = mgd = 1$.

3.2.1. Dormir

Consideremos que ya tenemos una capa de partículas que están dormidas y las cuales no pueden despertar bajo ninguna circunstancia. Lo que buscamos es si dejamos caer partículas normales sobre esta capa, al dormir las partículas se encuentren en posiciones estables mecánicamente. Esto, en un modelo de dinámica molecular es trivial: la sumatoria de fuerzas sobre la partícula debe ser nulo y encontrarse en un mínimo de la energía potencial. En el modelo ED, al no trabajar con el concepto de fuerza, debemos buscar condiciones en la evolución de la posición y la velocidad de la partícula para dormirla.

Condiciones físicas

En tres dimensiones una partícula necesita de, a lo menos, tres puntos donde apoyarse para estar en un equilibrio estable. En la medida que disipa energía sus rebotes se hacen cada vez más pequeños y más cercanos al punto de equilibrio. En el límite de infinitos choques, la partícula está en contacto con los puntos de apoyo y su velocidad es nula. Por lo tanto, debemos definir criterios para dormir que se cumplan un poco antes de que la partícula llegue a este punto, donde el colapso inelástico haría imposible la simulación.

Como no podemos encontrar el punto de manera exacta, debemos proponer una heurística: las condiciones siguientes se evalúan cada vez que una partícula normal choca con una sólida con una velocidad que no es capaz de despertar a la partícula dormida: a) que al chocar la rapidez de la partícula sea menor que v_{sleep} , b) que su trayectoria converja a un punto, c) que su velocidad este convergiendo a 0 y d) que la velocidad en z en aquel instante sea negativa. Si se cumplen estas 4 condiciones, la partícula que acaba de chocar se duerme. De las condiciones anteriores a) y b) son obvias, mientras que las otras dos las imponemos a fin de evitar que una partícula que esté “subiendo” sobre otra cambie de estado en el punto crítico donde la velocidad se anula más no las fuerzas actuando sobre ella. En la figura 3.2

se muestran cuatro configuraciones posibles y se explica si es factible o no dormir.

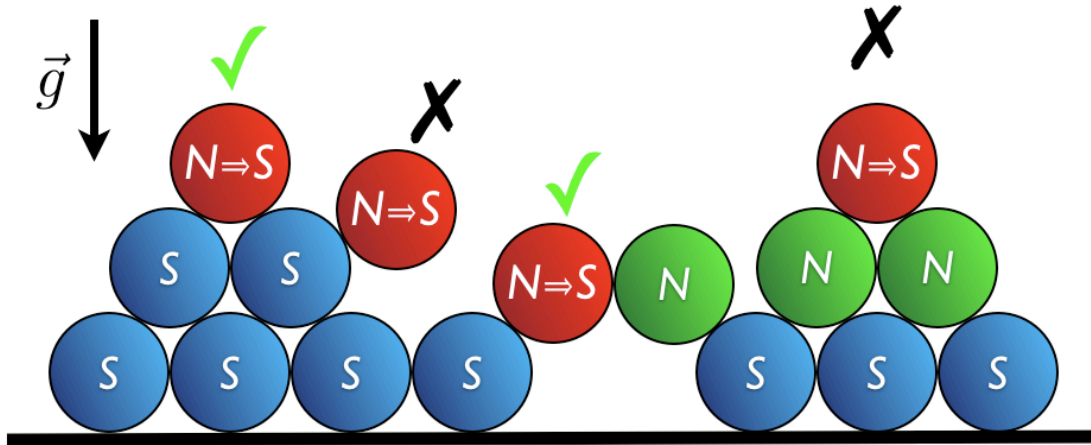


Figura 3.2: Esquema de las posiciones mecánicamente estable y no estables para dormir a las partículas en 2D. De izquierda a derecha: la primera partícula está en un punto estable, por lo que puede dormirse, la segunda está acelerando hacia abajo, por lo que no puede dormirse. La tercera está en una configuración estable gracias a la partícula normal y puede dormirse. La última partícula no puede dormirse puesto que no está en chocando con partículas sólidas.

Conviene notar aquí que no estamos seguros de que estas condiciones sean necesarias y suficientes para dormir a las partículas sólo en puntos estables. Sin embargo, no hemos encontrado configuraciones físicamente imposibles pero no las descartamos. En caso de que no fueran suficientes, existe la posibilidad de “despertar” a la partícula cada cierto tiempo con una velocidad igual a la que tenía cuando se durmió pero en una dirección cualquiera, de manera que si está en un punto estable, se quede ahí y si no, salga del punto crítico al cabo de un cierto tiempo.

3.2.2. Despertar

Las partículas pueden despertar en dos circunstancias. Al ser chocadas con suficiente energía o al perder la estabilidad mecánica.

En caso de ser chocada, debemos imponer una velocidad crítica para la partícula incidente v_{wakeup} , sobre la cual se despierte a la partícula dormida. Esto es necesario puesto que si $v_{\text{wakeup}} = 0$ estaríamos durmiendo y despertando a las partículas con cada choque, lo que haría tan ineficiente nuestra implementación como las anteriores.

En caso de perder la estabilidad mecánica, despertar a las partículas es un poco más complicado. Debido a que la simulación avanza de evento en evento y las partículas dormidas no son activas en los eventos que tienen, es decir, sólo son chocadas, debemos crear un mecanismo de verificación de la estabilidad. Llamamos “autodespertar” a este mecanismo y lo explicamos en detalle a continuación.

Autodespertar

Una configuración interesante para nuestro algoritmo es la estudiada por, Staron [19] en el colapso o desparramo (*spreading* en inglés) de columnas de granos. Partiendo del estado final de una deposición con gravedad, se sacan las paredes laterales y se estudia la geometría final del sistema con los parámetros que se muestran en la figura 3.3.

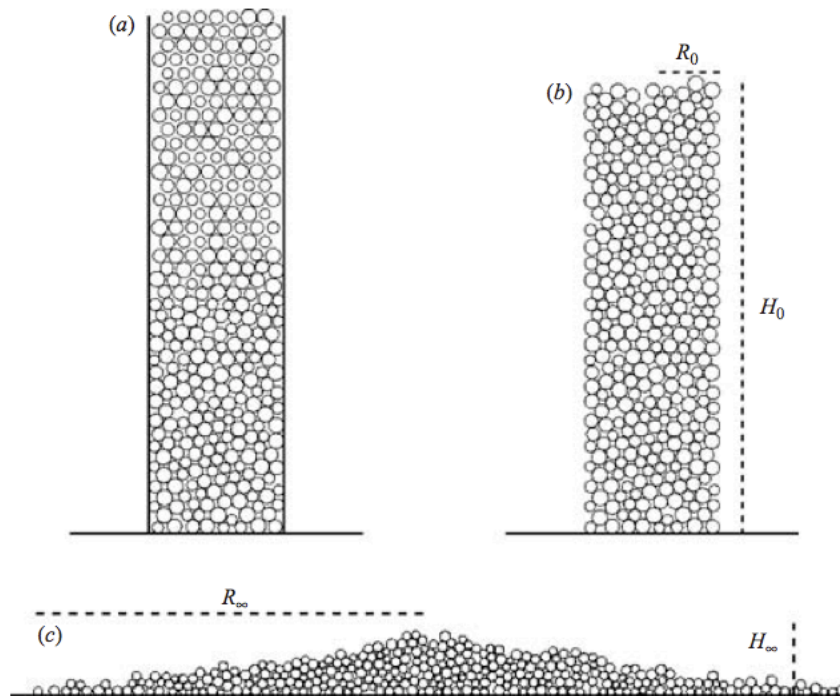


Figura 3.3: a) Preparación de una columna de granos por una lluvia aleatoria bajo la acción de la gravedad. b) La columna está caracterizada por su radio R_0 y altura inicial H_0 . c) Después del colapso, el depósito final está caracterizado por la distancia a la que llega el flujo R_∞ y su altura H_∞ . Figura tomada de [19].

Al realizar una deposición el estado final del sistema es uno con todas las partículas dormidas. Al quitar las paredes, las partículas de los bordes pierden sus puntos de apoyo y por lo tanto deben despertar sin que hayan sido chocadas por otras partículas.

Para dar cuenta de esto tuvimos que definir los “contactos” de una partícula dormida. Estas son las partículas sólidas que se encuentran a una distancia menor que R_c o las partículas normales que están a una distancia menor que R_c , se encuentran en el hemisferio superior (definido este por la dirección de la gravedad) de la partícula y que su rapidez sea menor que v_{sleep} . Como podemos ver en la figura 3.2 una partícula en 2D necesita de a la menos 2 puntos de apoyo. Estos pueden ser partículas dormidas o normales que estén a punto de dormirse, y por esto es que pedimos que su velocidad sea menor que v_{sleep} .

Gracias a la definición de contactos, llegamos a un algoritmo para autodespertar que cuenta con dos fases: las partículas están periódicamente comprobando que todos sus contactos cumplan aún las condiciones para ser contactos y sean más que el número mínimo de apoyos. Si no se cumple esto, se programa inmediatamente el despertar de la partícula. Esto hace que por un lado, no tengamos configuraciones físicamente imposibles por mucho tiempo, y por otro, que en sistemas con disipación y sin inyección de energía el sistema llegue efectivamente a un estado de energía nula al cabo de cierto tiempo debido a que no hay cambios en la disposición de los contactos.

El costo computacional de revisar si permanecen los contactos con los cuales se durmió la partícula es $\mathcal{O}(1)$: sólo necesitamos recorrer las celdas vecinas y calcular la distancia y el estado de las partículas que se encuentran en ellas y ver que todos los contactos han permanecido.

3.3. Implementación

En esta sección explicamos cómo las ideas anteriores se aplican a una simulación ED. Para esto, debemos explicitar como se atienden los eventos asociados al cambio de dinámica, de que manera se encuentran los tiempos de choque y cómo se realizarán estos.

3.3.1. Cambio de Estados

En esta sección explicaremos cómo realizar el cambio de estado de las partículas: puesto que nuestro sistema avanza por colisiones, el cambio de estado debe ser programado inteligentemente para que las partículas no queden en contacto.

Evento Dormir

Una vez que se ha decidido que una partícula debe dormirse (hecho esto en un evento de colisión contra un sólido, es decir, cuando están en contacto), debemos decidir cuando cambiar el estado de la partícula.

A primera vista, uno diría “cambiemoslo inmediatamente”. En el momento del choque las partículas se encuentran en contacto. El problema es que si la partícula es despertada en una colisión habrían tres partículas en contacto, y este choque para esferas duras no puede ser resuelto. Para evitar esto, debemos dormir a las partículas con una pequeña separación como se aprecia en la figura 3.1. Una vez decidido el cambio de estado de la partícula, calculamos los proximos eventos con la dinámica normal y agregamos uno antes que todos ellos: el evento dormir. Elejimos el tiempo de este evento como el menor entre $t_{sleep} = 10^{-5}$ (tiempo típico entre choques en una zona densa) y la mitad del tiempo para el proximo evento, de manera que al dormirse la partícula esté a lo más “a mitad de camino” entre la partícula con la que acaba de chocar y la del próximo choque.

La distancia típica a la que quedan separadas las partículas se puede acotar por $d_{tipico} \leq v_{sleep} t_{sleep}$ ya que la velocidad postcolisional de la partícula es menor que v_{sleep} y el tiempo de vuelo es menor o igual que t_{sleep} .

Evento Despertar

En caso que haya que autodespertarla una partícula, le damos un golpecito, es decir, una velocidad de dirección aleatoria y con una rapidez igual a v_{sleep} y cambiamos su estado a normal.

Como las partículas se duermen separadas al momento de despertarlas por un choque el cambio de dinámica es trivial: se atiende el choque como si fueran dos partículas normales, pero con un extra de energía (si corresponde) y se cambia el estado de la partícula a normal.

La dinámica de las partículas dormidas es, por cierto, nula, pero debemos conservar o mejor dicho, disipar a una tasa adecuada la energía del sistema. Esto se comprende volviendo a la figura 3.1, al dormirla ántes de t^* perdemos energía a una tasa mayor que la real. Esto hace que la densidad local aumente, lo que facilita el colapso inelástico. Por lo tanto, debemos buscar una forma para despertar a las partículas de manera que en el intervalo $\Delta t = t^* - t_{sleep}$ la energía se conserve.

La solución que hemos escogido para sobrellevar esto es guardar la energía cinética que tenía la partícula al dormirse y liberarla en el choque que la despierta. Esto es debido a que el proceso de dormir a las . De esta manera, hacemos que la tasa de disipación disminuya y por ende la densidad quede constante. Al implementar esta recuperación de energía, hemos elegido agregar la energía como un extra a la componente normal de la velocidad:

$$\delta = \frac{1}{2}(V_{cn} + \sqrt{V_{cn}^2 + 4E_{Sleep}}), \quad (3.1)$$

donde δ es la velocidad normal relativa después del choque y E_{Sleep} la energía cinética de la partícula al momento de dormirse. Esta energía la guardamos solo durante un tiempo t_{check} puesto que eventualmente las partículas reales pierden toda la energía.

Antes de comprender el rol que jugaba la tasa de pérdida de energía jugamos con otras formas de despertar/dormir que detallamos a continuación:

La idea inicial fue dormir a las partículas con una velocidad nula y despertarlas si la partícula incidente tenía una velocidad mayor que cierta cota. El choque lo atendíamos como un choque normal donde una de las partículas tenía velocidad nula. Para que el sistema no colapsase debíamos imponer una cota para la velocidad de la partícula incidente bastante grande, del orden de la unidad.

Luego de estudiar muchos casos en los que el colapso se producía, decidimos que al dormir las partículas solo pueden despertar pasado un cierto tiempo. Esto es porque los colapsos se producían cuando las partículas estaban en contacto constante, una sobre la otra “resbalando”.

Ninguna de estas soluciones nos permitió alcanzar cotas para despertar muy bajas, ya que seguíamos perdiendo energía a una tasa muy pequeña.

3.3.2. Predicción de Eventos

La predicción de eventos consiste en calcular el tiempo en que una partícula o bien choca con otra o choca con una pared. Para calcular el tiempo de choque con paredes quietas las ecuaciones a resolver son a lo más cuadráticas (dependiendo si hay o no gravedad en la dirección de la pared) y están dadas por la cinemática de la partícula. En caso de paredes móviles las ecuaciones se complican y no las consideraremos en este trabajo.

En el caso en que las dos partículas sientan la misma gravedad, la ecuación a resolver es cuadrática y su implementación ya ha sido discutida en otras partes [23]. En la figura

3.4 se ve que la si ambas partículas sienten la misma gravedad, su movimiento se puede ver como con velocidad constante, de esta manera, dos partículas pueden o bien no chocar nunca, y la ecuación tiene cero soluciones reales o intersectarse dos veces, de entrada y de salida. El tiempo físico que nosotros buscamos es el en que la velocidad y la normal entre las dos partículas es negativo, es decir, una partícula va “entrando” en la otra.

Conviene notar en este punto que si dos partículas normales acaban de chocar, es imposible que choquen nuevamente sin que alguna de ellas cambie su estado antes debido a otro choque. Esto es porque al chocar, ambas partículas se están alejando y continúan así a menos que una choque y modifique su velocidad.

La predicción de choques entre partículas con distintas dinámicas es algo que presenta ciertas dificultades numéricas que debemos sobrellevar. Debido a que ahora hay partículas que no son afectadas por la gravedad, obtenemos ecuaciones de mayor grado. Los tiempos de choque se calculan al imponer que el módulo de la diferencia de las posiciones de las partículas sea igual a la suma de los radiosap:

$$\|\vec{r}_i - \vec{r}_j\| = R_i + R_j \quad (3.2)$$

Elevando al cuadrado esta ecuación, se obtiene el polinomio que entrega el tiempo del próximo choque. Este polinomio puede ser de orden 4,3 ó 2, dependiendo del estado de las partículas involucradas y el momento en que se calcula.

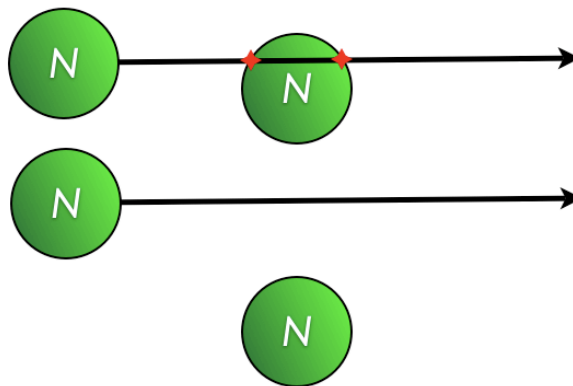


Figura 3.4: Esquema de las dos posibles situaciones al predecir el tiempo de choque entre partículas normales, la flecha representa la trayectoria de la partícula izquierda en el sistema de referencia de la partícula derecha. En el caso superior, las partículas se intersectan dos veces (estrellas rojas) y en el caso inferior no hay intersección.

Los polinomios de orden 4 aparecen si es que una de las partículas siente la gravedad y la otra no. Este es el caso, por ejemplo, de la predicción entre una partícula sólida y una normal. En este caso, la partícula normal puede pasar 4 veces por la partícula dormida, dos veces en “subida” y dos en “bajada”. En la figura 3.5 se muestra la trayectoria del centro de masa de una partícula y tres maneras de intersectar el volumen excluido de la partícula dormida. La ecuación a resolver en este caso es:

$$\frac{\vec{g}^2}{4}t^4 + \vec{g} \cdot \Delta\vec{v}t^3 + (\Delta\vec{v}^2 + \vec{g} \cdot \Delta\vec{r})t^2 + 2\Delta\vec{v} \cdot \Delta\vec{r}t + \Delta\vec{r}^2 - \sigma^2 = 0 \quad (3.3)$$

Hemos definido $\Delta\vec{v} = \vec{v}_1 - \vec{v}_2$ y $\Delta\vec{r} = \vec{r}_1 - \vec{r}_2$, las diferencias de las velocidades y posiciones de las partículas entre las cuales se predice el tiempo de choque.

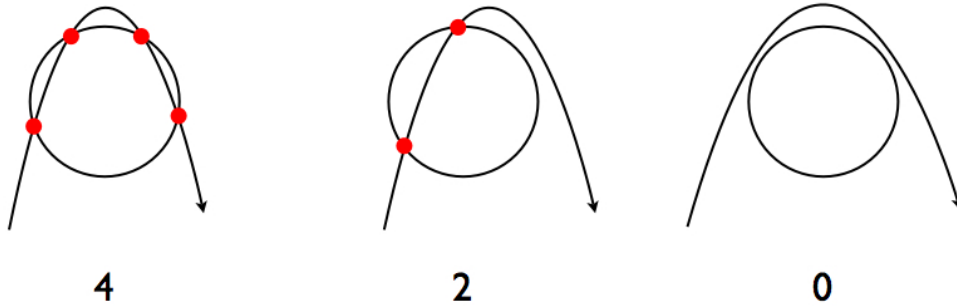


Figura 3.5: Diagrama de los tres posibles casos al predecir los tiempos entre partículas con distinta gravedad efectiva: la ecuación puede tener 4, 2 ó 0 soluciones reales, que corresponden a las tres formas posibles de intersectar una parábola con un círculo.

La ecuación anterior es la más general que podemos escribir y puede ser simplificada dependiendo del momento en que se realice la predicción.

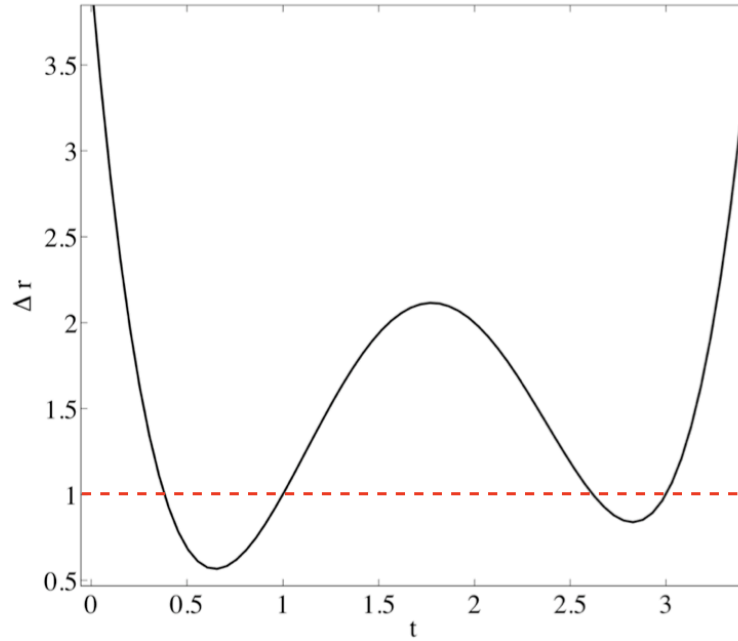


Figura 3.6: Distancia entre los centros de masa para una partícula sólida y una normal como función del tiempo, que corresponde al primer caso de la figura 3.5. El polinomio entrega 4 soluciones que son los instantes en que entra y sale de la otra partícula. Las intersecciones de la línea sólida con la punteada indican las cuatro soluciones a la ecuación 3.3

Cuando dos partículas acaban de chocar se cumple que $\Delta \vec{r}^2 = \sigma^2$. Luego simplificando un t en la ecuación 3.3, el tiempo de choque entre esas dos partículas tiene la siguiente ecuación de tercer grado.

$$\frac{\vec{g}^2}{4}t^3 + \vec{g} \cdot \Delta \vec{v}t^2 + (\Delta \vec{v}^2 + \vec{g} \cdot \Delta \vec{r})t + 2\Delta \vec{v} \cdot \Delta \vec{r} = 0 \quad (3.4)$$

Solución Numerica

La solución de la la ecuación 3.4 la encontramos con un método de Dave Eberly [36]. Para resolver la ecuación 3.3 debemos encontrar las soluciones de manera iterativa puesto que el enfoque analítico es inestable y conduce a una mala obtención de raíces.

Método de Derivada y Bisección

Como el problema numérico que tenemos al resolver la ecuación 3.3 es que el coeficiente constante del polinomio es muy pequeño, la solución que necesitamos no puede depender de él. Para eso derivamos la ecuación 3.3 obteniendo una ecuación para los mínimos de la distancia (ecuación 3.5). De esta manera obtenemos una ecuación que no depende del término constante $\Delta \vec{r}^2 - \sigma^2$.

$$4\frac{\vec{g}^2}{4}t^3 + 3\vec{g} \cdot \Delta\vec{v}t^2 + 2(\Delta\vec{v}^2 + \vec{g} \cdot \Delta\vec{r})t + 2\Delta\vec{v} \cdot \Delta\vec{r} = 0 \quad (3.5)$$

Como se ve en la figura 3.6, la distancia entre las partículas tiene 3 extremos, de los cuales a nosotros sólo nos interesan el primero y el tercero. Si alguno de estos mínimos se encuentra en el futuro, debemos evaluar la distancia entre las partículas en ese instante. En caso de que estén traslapadas, sabemos que la solución a la ecuación 3.3 que nos interesa se debe encontrar entre el tiempo actual y el tiempo de la mínima distancia. De esta forma, gracias al método de bisección encontramos numéricamente, a la precisión que nosotros queramos, el tiempo de choque asegurándonos de que las partículas no estén traslapadas.

3.3.3. Choques

En esta sección describimos los cambios que se deben hacer a las reglas de choques debido al cambio de estado de las partículas. Para esferas duras inelástica sin rotación:

$$\vec{v}'_1 + \vec{v}'_2 = \vec{v}_1 + \vec{v}_2 \quad (3.6)$$

$$\vec{v}'_{21n} = -\alpha\vec{v}_{21n} \quad (3.7)$$

$$\vec{v}'_{21t} = \vec{v}_{21t} \quad (3.8)$$

Donde las ecuaciones representan la conservación del momentum, la inelasticidad del choque (dado por el coeficiente de restitución α) y la ausencia de roce respectivamente.

Por otra parte, en el caso en que una de las partículas se encuentre en estado sólido, el choque es como contra una pared, es decir:

$$\vec{v}'_{1n} = -\alpha_s\vec{v}_{1n} \quad (3.9)$$

$$\vec{v}'_{1t} = \vec{v}_{1t} \quad (3.10)$$

la componente normal del choque se invierte y se disipa con un factor α_s mientras que la componente tangencial se mantiene.

3.4. Constantes introducidas

A continuación presentamos todas las constantes que hemos introducido a fin de implementar las ideas anteriormente dichas.

Constante	Valor	Definición
v_{sleep}	0,1 – 0,01	Rapidez mínima para dormir
v_{WakeUp}	0,1 – 0,01	Rapidez mínima para despertar
t_{check}	0,1 – ∞	Tiempo característico para revisar los contactos
t_{sleep}	10^{-4}	Máximo tiempo para atender evento dormir
α_s	0,4	coef. rest. con sólido

En la medida en que las cotas para dormir y despertar se hacen más chicas, el número de choques aumenta haciendo que la simulación sea más lenta. Recomendamos usar cotas del orden de $0,05\sqrt{gd}$ para realizar mediciones. En caso de hacer vídeos, conviene poner las cotas del orden del diez por ciento ya que hacen considerablemente más rápida la simulación y los resultados son cualitativamente similares.

En los casos de estudio de los siguientes capítulos mantendremos constante todos los parámetros salvo el tiempo de revisión y las velocidades para dormir y despertar. El tiempo de revisión lo impondremos infinito para la deposición mientras que para el tambor rotante y la avalancha usaremos un $t_{\text{check}} = 0,1$. Estudiaremos cómo dependen las propiedades macroscópicas de nuestros sistemas al variar estas velocidades y encontraremos cuáles son los valores que, por un lado son eficientes computacionalmente, y por otro dan la física adecuada.

3.5. Áreas de Aplicación y Limitaciones

Una pregunta que surge de inmediato es si nuestra extensión evita el colapso inelástico. No lo hace por completo. Aun más, no es útil para la configuración donde se encontró el colapso inelástico debido a que la hemos asumido la presencia de gravedad para definir los criterios bajo los cuales dormir a las partículas.

Otra limitación de nuestra extensión es la falta de rotación y roce de nuestras partículas. Generalizar los criterios para dormir y despertar sin embargo es directo: basta con agregar a las condiciones para dormir una cota a la rotación de la partícula.

Nuestro algoritmo está pensado para utilizarse en sistemas que posean una fase sólida o estancada, ya que es en estos casos donde la eficiencia computacional es considerable y vale la pena modificar el método ED estándar.

Un ejemplo fácil son estudios de estructuras de empaquetamiento de esferas, donde se busca precisamente el estado con energía cinética nula. Por ejemplo, nuestro método puede

ser usado para estudiar compactación de medios granulares [25].

El caso de enfriamiento de gases en ausencia de gravedad donde se forma conglomerados [27] no puede ser resuelto por nuestra extensión. Aun así, pensamos que la idea más general de usar distintas dinámicas en la misma simulación puede ser aplicado para describir bajo una misma dinámica a las partículas que pertenezcan a un mismo conglomerado.

En los flujos granulares, área de gran interés teórico y práctico [35], nuestro algoritmo es útil en flujos cuyo perfil de velocidad no es lineal, es decir, que presenten una fase atascada, donde las escalas de tiempo del movimiento son mucho mayores a las del flujo de la superficie. Este es el caso del tambor rotante y el flujo en pilas de arena por ejemplo.

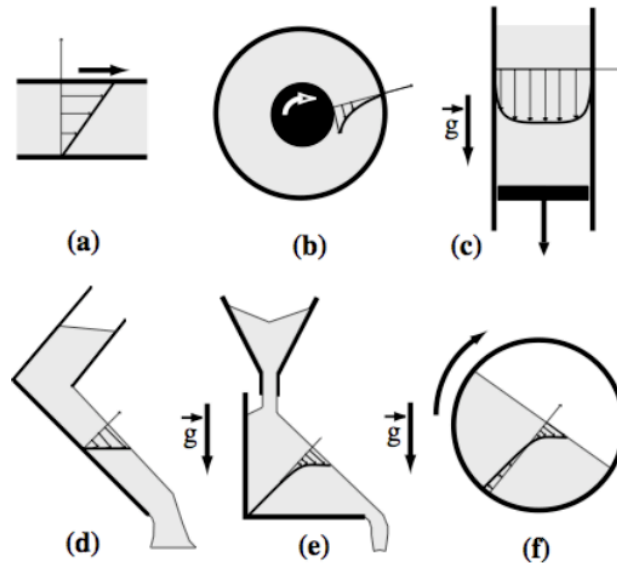


Figura 3.7: Diferentes configuraciones experimentales de flujos granulares densos. Nuestro algoritmo es de utilidad en b), c), e) y f) donde los perfiles de velocidad presentan una cola exponencial. Diagrama tomado de la referencia [35].

En los capítulos siguientes aplicaremos este algoritmo a distintos sistemas nunca antes accesibles a simulaciones ED.

Capítulo 4

Deposición

En este capítulo presentamos los resultados que obtenidos al estudiar la deposición uniforme de partículas sobre un suelo estático. Este caso nos permite estudiar el efecto del cambio las velocidades para dormir y despertar en la física de un sistema y estudiar la eficiencia computacional de nuestro modelo.

4.1. Introducción

El empaquetamiento de esferas iguales ha sido estudiado por biólogos, ingenieros, químicos y físicos para entender la estructura de células vivas, líquidos, medios granulares y vidrios, para mencionar algunos ejemplos [15].

Un empaquetamiento ordenado (FCC o HCP) de esferas en 3D alcanza un fracción de volumen máxima de

$$\phi_{\text{Kepler}} \simeq 0,74048 \tag{4.1}$$

Este valor es considerablemente mayor que el de random close packing $\phi_{\text{rcp}} = 0,64$ encontrado por Torquato et al. [33], que está definido como la mayor densidad posible en empaquetamientos al azar. Si se quiere comprimir más el random close packing, necesariamente deben organizarse las partículas en estructuras ordenadas, presentando una transición de fase entre los dos estados.

En el otro extremo se encuentra el “random loose packing” $\phi_{\text{rlp}} = 0,52$ que es la menor densidad posible que mantiene la rigidez del sistema. Para que esto suceda, cada partícula debe estar en contacto con a lo menos otras cuatro, y los cuatro puntos de contacto no pueden estar en el mismo hemisferio o en el mismo ecuador.

Como primer caso de estudio de nuestro algoritmo, estudiaremos la fracción de empaquetamiento como función de las cotas para dormir y despertarse. Esperamos dos cosas de este experimento. Por un lado, que el valor de nuestro ϕ sea cercano a ϕ_{rec} debido al protocolo de deposición que estamos ocupando: al depositar con una densidad baja y a velocidad pequeña damos tiempo a que las partículas encuentren el mejor punto de equilibrio y a la vez generen reacomodos en la superficie de la deposición. Por otro, que exista una convergencia de la fracción de empaquetamiento a medida que las velocidades para dormir y despertar se acerquen a cero recuperando la simulación original ED, debido a que al dormir a menor velocidad las partículas quedan con una separación menor.

4.2. Configuración

Tenemos una caja con condiciones de borde periódicas de $10d \times 10d$ cuyo suelo es una configuración aleatoria de 200 partículas sólidas que no despiertan, para evitar capas de ordenamiento cristalino en el principio de la deposición. En la figura 4.1 se aprecia el sistema en tres instantes de tiempo.

El tiempo de verificación lo fijamos $t_{check} = \infty$ dado que en una deposición una partícula que queda en una posición estable no pierde sus contactos nunca más.

El resto de los parámetros de la simulación están fijos e iguales a los presentados en 3.4 salvo $v = v_{sleep} = v_{WakeUp}$ que variamos entre 0,01 y 0,1.

4.3. Protocolo

La idea es hacer “llover” partículas sobre el suelo de manera aleatoria y con una velocidad pequeña para que la configuración final obtenida sea desordenada. En concreto dejamos caer 4800 partículas con una distribución uniforme de velocidades ($v_z \sim -1,2$, equivalente caer $0,7d$ bajo la acción de la gravedad) en capas con una fracción de empaquetamiento inicial cercana al $\phi_{ini} \sim 0,04$. Las partículas se mueven con velocidad constante en z hasta que chocan por primera vez. Esto emula una configuración experimental en que un reservorio se levanta con la misma velocidad que la pila crece, de manera que las partículas siempre lleguen con la misma velocidad a la superficie de la deposición.

Dejamos evolucionar el sistema por 600 unidades de tiempo, suficiente para que todas las partículas alcancen su punto de equilibrio. Para obtener la fracción de empaquetamiento

ordenamos las partículas por su altura y calculamos el número de partículas en función de la altura en la zona central de la deposición. Con esto, hacemos una regresión lineal y la pendiente de la recta que las relaciona es directamente proporcional a la fracción de empaquetamiento.

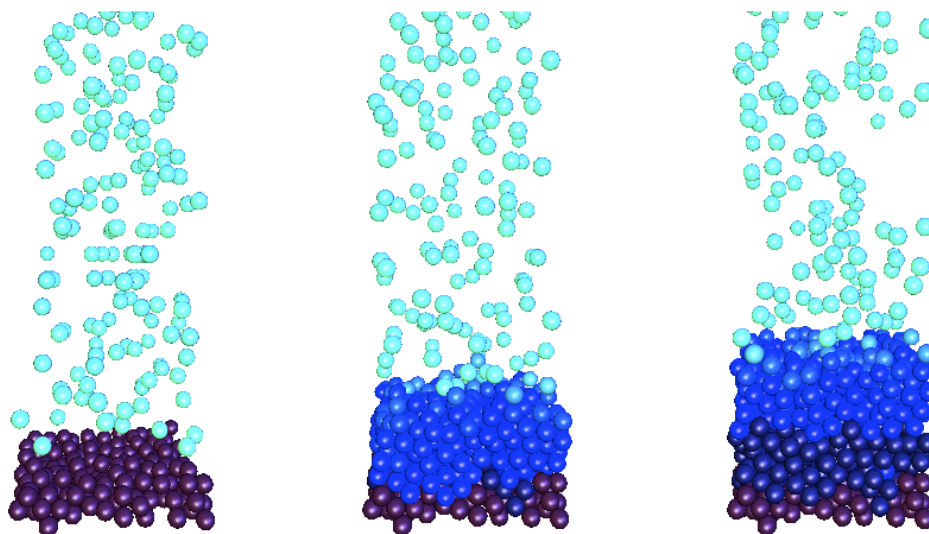


Figura 4.1: Foto para los tiempos $t = 0\sqrt{d/g}; 50\sqrt{d/g}; 100\sqrt{d/g}$ (ordenados de izquierda a derecha) de una de nuestras deposiciones. Las partículas moradas pertenecen al suelo estático. Los distintos tonos de azul dependen de la energía cinética de las partículas mientras que el azul oscuro indica las dormidas.

4.4. Resultados

Realizamos 100 deposiciones para v entre 0,01 y 0,1. Todas las simulaciones entregaron una fracción de empaquetamiento dentro de lo esperado de acuerdo a la literatura, variando tan solo en un 1%. Con estos valores (figura 4.2) realizamos el siguiente ajuste

$$\phi = 0,640(\pm 0,0005) - 0,10(\pm 0,007)v \quad (4.2)$$

y encontramos que el valor asintótico de $\phi_0 = 0,640$ igual al random close packing ($\phi_{RCP} = 0,640$) encontrado en la literatura [33].

Interpretamos esto como que nuestro algoritmo está durmiendo las partículas en posiciones físicamente correctas. Si no lo hiciera, el empaquetamiento debería ser menor, que es lo que pasa cuando aumentamos las cotas: al aumentar la velocidad en la cual se duermen las partículas, no disipamos suficiente energía y la densidad disminuye.

Eficiencia

Se estudió el número de eventos totales de la simulación, lo que da una estimación del tiempo computacional usado (figura 4.3). Se encontró que el número de eventos diverge a medida que disminuimos v con el ajuste $N_{\text{eventos}} \sim v^{-1,2}$. La razón es que cada partícula tiene más eventos antes de dormirse, y en el límite en que no se duermen, el sistema nunca alcanza un estado sin energía cinética, que es el requisito para terminar la simulación.

Estudiamos el tiempo de CPU necesario para simular un cierto tiempo en la simulación. Los resultados se aprecian en la figura 4.5. Como nuestra extensión duerme las partículas simulamos un número efectivo de ellas mucho menor al del sistema total y que se mantiene prácticamente constante durante la simulación. En la figura 4.4 se muestra el número de partículas en cada estado para una deposición con $v = 0,01\sqrt{gd}$. Nuestro algoritmo simula a lo más el 20% de las partículas del sistema, mientras que si no las durmieramos deberíamos simularlas todas, lo que haría que el tiempo total de la simulación aumentara con $\mathcal{O}(N^2)$.

Debido a que sólo gastamos tiempo computacional en las partículas que están en la superficie de la deposición, nuestro algoritmo tiene un costo constante en el tiempo, mientras que los anteriores, al simular todas las partículas en la deposición, hace que el costo computacional crezca a medida que más partículas se van agregando. De esta manera, el costo computacional como función del tiempo físico de la simulación es para nuestra extensión $\mathcal{O}(t)$ mientras que para una ED convencional, el costo computacional crece como $\mathcal{O}(t^2)$

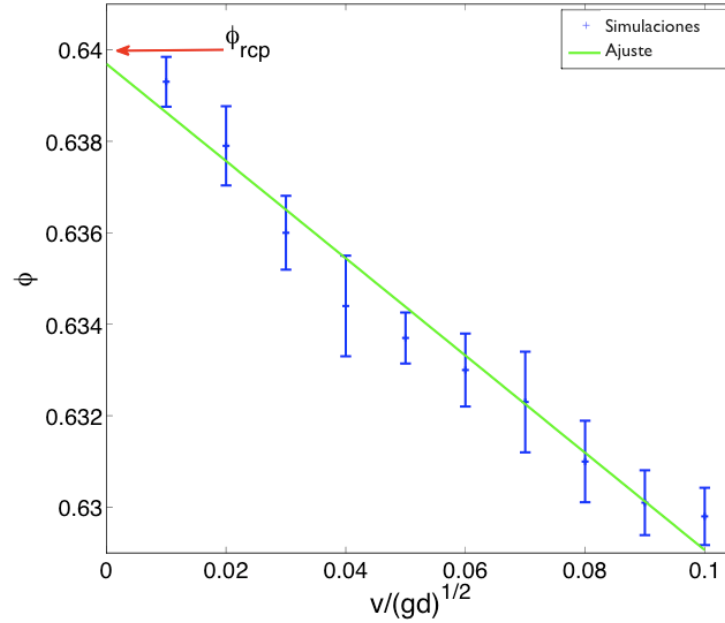


Figura 4.2: Fracción de empaquetamiento como función de las cotas para dormir y despertar. El valor del observable físico varía en un 1% mientras nuestras cotas varían un orden de magnitud.

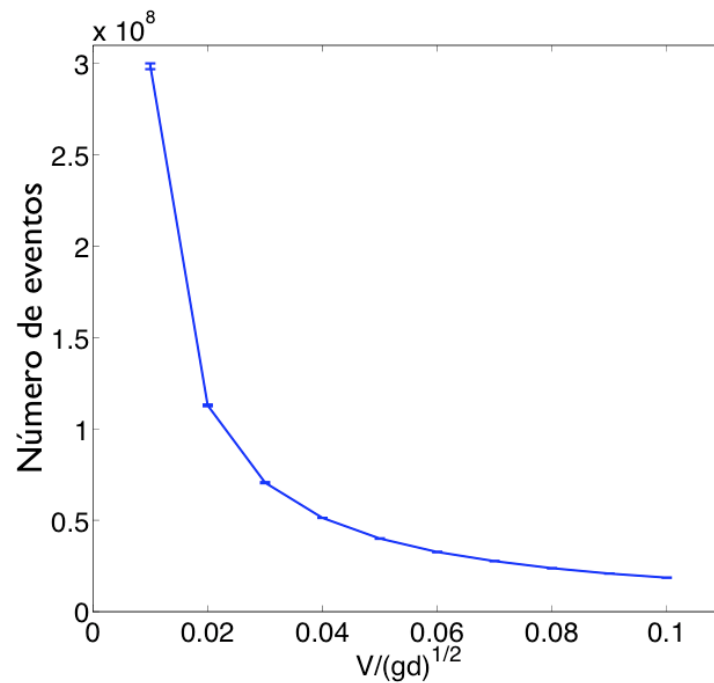


Figura 4.3: Costo computacional para deposiciones. En el eje horizontal las cotas para dormir y despertar y en el eje vertical el número de eventos totales de la simulación.

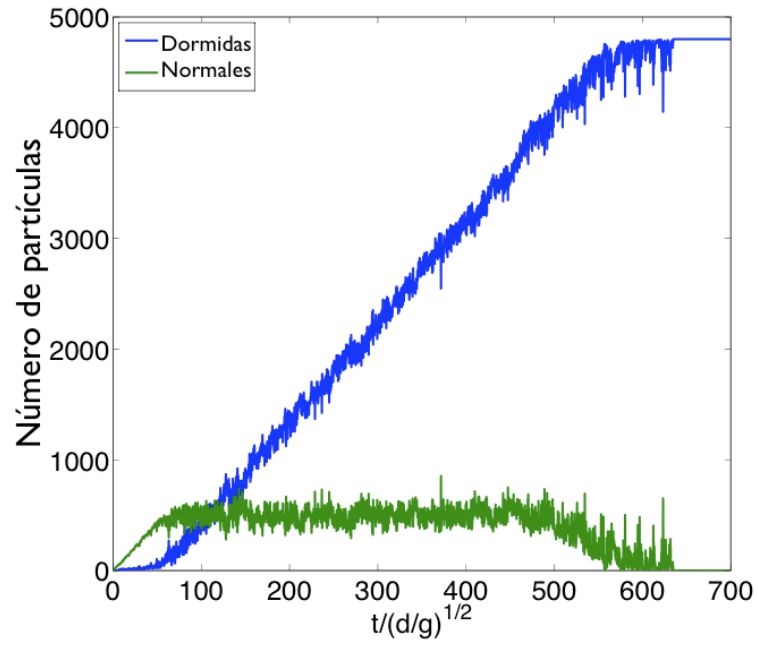


Figura 4.4: En el eje horizontal el tiempo y en el eje vertical el número de partículas en cada estado. Se aprecia que el número de partículas normales permanece acotado.

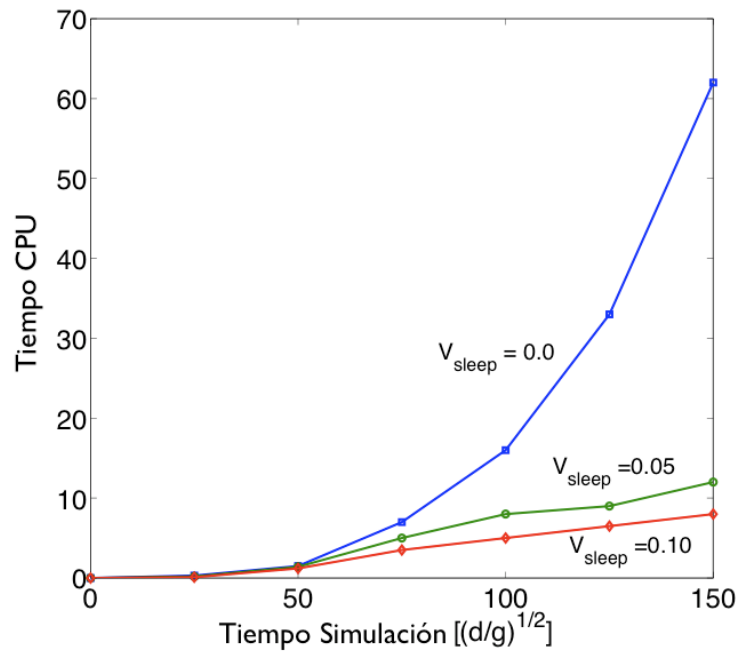


Figura 4.5: En el eje horizontal el tiempo de la simulación y en el eje vertical el tiempo de calculo en minutos. Como se aprecia, si no se duermen las partículas, el tiempo aumenta cuadráticamente.

4.5. Proyecciones

Como hemos visto, nuestro algoritmo es capaz de simular empaquetamientos de esferas de una manera eficiente. Estudios futuros podrían incluir diferentes protocolos de deposición, a fin de obtener distintas fracciones de empaquetamiento, o estudiar de que manera compactificar un empaquetamiento de esferas: golpenando el suelo o las paredes, cual es la amplitud óptima de los golpes, etc.

Capítulo 5

Avalancha gruesa

Por avalancha gruesa entendemos los flujos de material granular sobre planos inclinados. Un ejemplo de esto en la naturaleza son los derrumbes de tierra en las laderas de las montañas.

En este capítulo presentamos los resultados obtenidos al estudiar el ángulo de detención de flujos densos sobre un plano inclinado rugoso. Los resultados muestran un comportamiento cualitativamente similar a experimentos y simulaciones anteriores.

5.1. Introducción

Luego de haber estudiado la fracción de empaquetamiento para una deposición queremos ver de qué manera las propiedades de un sistema dinámico dependen de los parámetros de nuestro algoritmo. Para tal efecto, estudiaremos el ángulo de detención para avalanchas de distintas alturas en planos inclinados rugosos. En la medida que la altura de la avalancha aumenta, le es más fácil fluir y le cuesta más detenerse. La relación entre el ángulo de reposo y la altura h de una avalancha se conoce en la literatura como la curva $\theta_{\text{stop}}(h)$ y es de gran importancia para la reología de flujos granulares densos.

Las avalanchas en planos inclinados son una de las configuraciones más estudiadas, tanto experimental como teórica y simulacionalmente. Ejemplos de estos flujos en la naturaleza son las avalanchas de rocas en las montañas y deslizamientos de arena en las dunas. Los flujos densos de granos macroscópicos presentan fenómenos que son comúnmente encontrados en sólidos amorfos o enfriados, como *jamming* y transición de vidrio, cristalización y transición de primer orden de estados desordenados a ordenados [10].

Para flujos estacionarios una de las propiedades más asombrosas que presentan es que es posible colapsar la velocidad media del flujo como función de la altura para distintas configuraciones, al ser debidamente escalada la altura por h_{stop} mostrando un comportamiento

universal. Pouliquen [35] encontró la siguiente relación empírica:

$$\frac{\langle v \rangle}{\sqrt{gd}} = A + B \frac{h}{h_{\text{stop}}} \quad (5.1)$$

donde A y B son constantes para cada experimento, mostrando que la influencia de la inclinación está codificada en h_{stop} (figura 5.1).

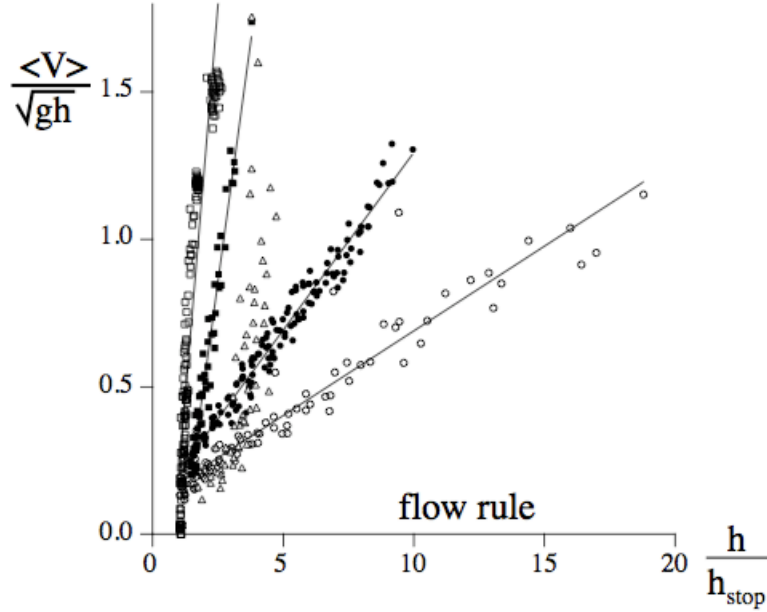


Figura 5.1: El número de Froude $\langle v \rangle / \sqrt{gd}$ en función de altura escalada por h_{stop} para distintas configuraciones experimentales, tomado de [35] .

5.2. Configuración

Experimentalmente se controla el flujo de partículas para un ángulo fijo, hasta que la altura sea tal que no haya flujo en montajes como el de la figura 5.2. De esta manera para cada ángulo se obtiene la altura a la cual se detiene el flujo, es decir, $h_{\text{stop}}(\theta)$.

En nuestro caso, es más difícil sacar partículas de la simulación que mover el ángulo de inclinación y por esto elegimos la configuración de la derecha en la figura 5.2 obteniendo $\theta_{\text{stop}}(h)$. En experimentos anteriores se ha visto que estas curvas son una la inversa de la otra, por lo que ambas configuraciones son equivalentes.

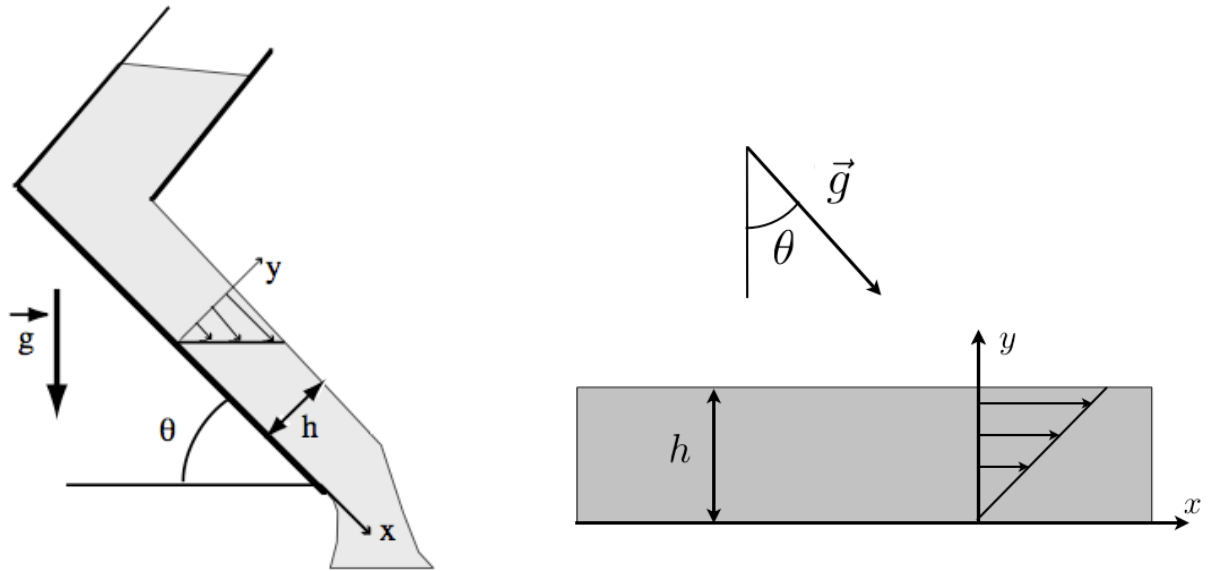


Figura 5.2: Izquierda: Configuración experimental para estudiar h_{stop} . El flujo se regula por la altura de salida para el contenedor [35]. Derecha: Configuración de nuestras simulaciones para estudiar θ_{stop} . Una vez fijado el número de partículas en el sistema, el flujo se regula por el ángulo que está dado por la dirección de la gravedad. Hay condiciones de borde periódicas en x .

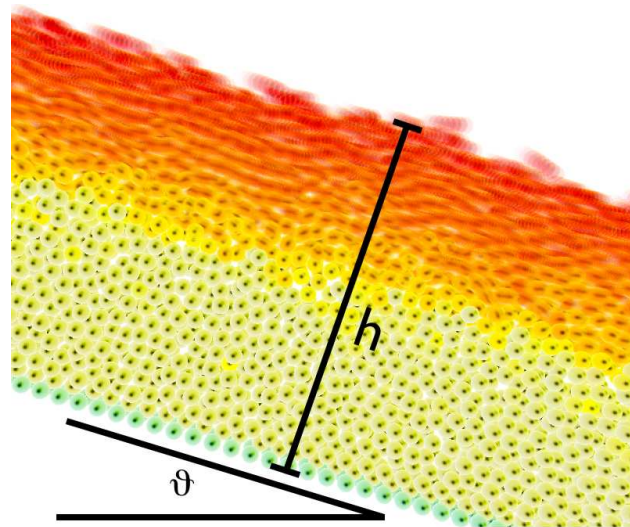


Figura 5.3: Foto con el obturador abierto para una avalancha de 10000 partículas en una cama de 40×10 diámetros.

Usamos los mismos parámetros que en las simulaciones de deposición fijando el valor de $V_{\text{sleep}} = V_{\text{wakeUp}} = 0,05$ que basados en los resultados de las deposiciones nos parece un valor razonable. Sobre un suelo rugoso de $40d \times 10d$ de área hecho con ~ 2000 partículas que no pueden despertar ordenadas en una grilla cuadrada, hacemos simulaciones con un

número variable de partículas entre 500 y 10000 y ángulos entre 10° y 20° . El sistema tiene condiciones de borde periódicas en x e y y es la gravedad la encargada de fijar el ángulo.

Para obtener un flujo estacionario fijamos θ y dejamos caer las partículas sobre el suelo hasta que se desarrolle el estado estacionario. Consideramos que este se alcanza cuando la energía del sistema alcanza un valor constante en torno al cual fluctúa. El tiempo en alcanzar el estado estacionario depende del número de partículas en el sistema y para el caso de 10000 partículas es cercano a los $10000\sqrt{g/d}$.

Una vez obtenido el régimen estacionario procedimos a disminuir el ángulo a una tasa $\dot{\theta} = \frac{\delta\theta}{\delta t}$ constante. Como nuestro modelo considera que la gravedad es constante, debemos cambiar el ángulo en pequeños escalones $\delta\theta$. La disminución del ángulo la realizamos en pasos discretos cada $\delta t = 0,2\sqrt{g/d}$. Para modificar $\dot{\theta}$ cambiamos el $\delta\theta$ en un rango de valores, siempre teniendo en cuenta de que el cambio de ángulo sea lento comparado a la escala de tiempo del sistema, representado por el tiempo de relajación.

5.3. Resultados

Dependencia de θ_{stop} en $\dot{\theta}$

Antes de obtener la curva $\theta_{stop}(h)$ debemos buscar cual es la manera correcta de disminuir el ángulo de la rampa. Debido a que el sistema tiene un tiempo de relajación característico, si la tasa de cambio de ángulo $\dot{\theta}$ es muy grande obtenemos que la detención se realiza lejos del ángulo crítico. Esta prueba es importante puesto que el fenómeno que estamos estudiando presenta una región metaestable. Para obtener con certeza este ángulo deberíamos disminuir el ángulo de la avalancha adiabáticamente, como la velocidad depende del ángulo esto hace que la simulación sea extremadamente costosa.

El ángulo de detención es fácilmente identificable para $\dot{\theta}$ pequeño, puesto que la energía cinética promedio del sistema tiene un salto abrupto como se puede ver en la figura 5.4 para la curva azul, que es la con menor $\dot{\theta}$. En la figura 5.5 se puede ver la evolución de la energía cinética promedio en función del tiempo, mostrando el tiempo de simulación necesario para simular los distintos valores de $\dot{\theta}$.

Para estudiar esta dependencia, realizamos 5 simulaciones (figura 5.4) con $\dot{\theta}$ en un rango de tres órdenes de magnitud obteniendo que el ángulo de detención depende de $\dot{\theta}$: entre menor es $\dot{\theta}$ mayor es el ángulo de detención, tendiendo a un valor finito. Usamos los siguientes valores

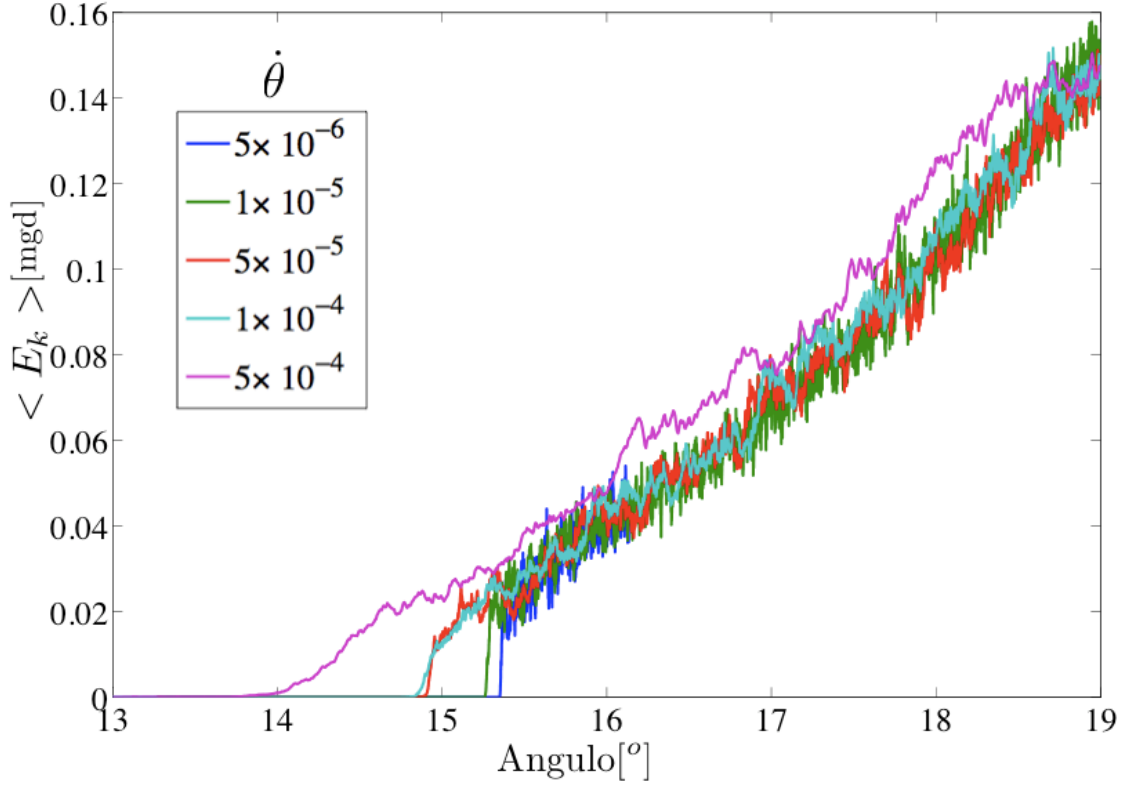


Figura 5.4: Evolución de la energía cinética promedio $\langle E_k \rangle = \frac{1}{2N} \sum_i m v_i^2$ en función del ángulo para una avalancha de $N = 2000$ partículas mientras se disminuye el ángulo a distintos $\dot{\theta}$.

para $\dot{\theta} : (5 \times 10^{-6}, 10^{-5}, 5 \times 10^{-5}, 10^{-4}, 5 \times 10^{-4})$.

Debido a que es necesario terminar las simulaciones de avalanchas en un tiempo razonable, usamos $\dot{\theta} = 1 \times 10^{-4}$ que fuera razonable de simular y que el error en la determinación del ángulo sea comparable a los resultados experimentales (del orden de un grado). Con el valor elegido, la simulación de un sistema con 10000 partículas toma una semana de calculo.

Determinación de θ_{stop} para nuestras simulaciones

Cualitativamente hemos recuperado la forma de la curva $\theta_{stop}(h)$ que se observa tanto en experimentos como en simulaciones para las avalanchas. Para obtener el valor del ángulo de detención y la altura de la avalancha procedimos de la siguiente forma: definimos θ_{stop} como el ángulo cuando la energía cinética de las partículas era menor que el 1% de la energía típica de una partícula, $E_{típica} = mgd$. Luego, tomamos la configuración final de la avalancha y medimos la altura promedio de las partículas en la superficie, para esto, dividimos el plano en celdas verticales de $1,5d \times 1,5d$ y buscamos la partícula a mayor altura en esa celda.

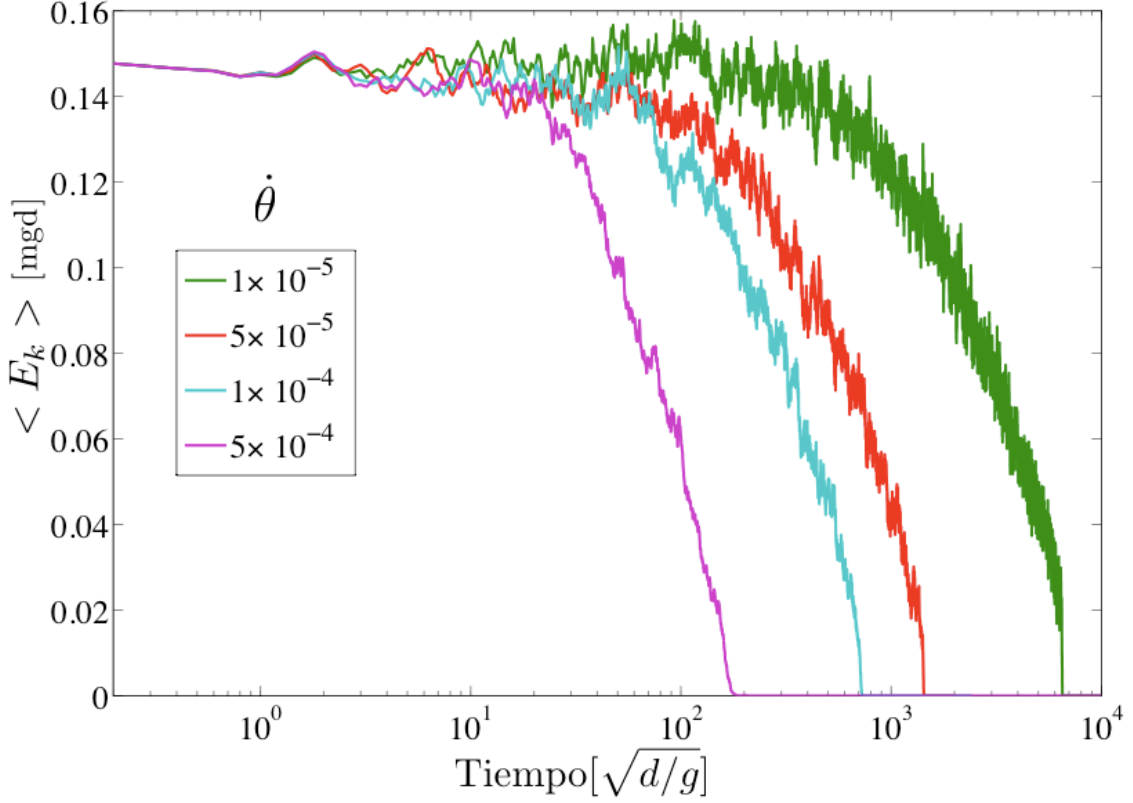


Figura 5.5: Evolución de la energía para una avalancha de 2000 partículas mientras se disminuye el ángulo a distintos $\dot{\theta}$, esta vez graficado en función del tiempo de la simulación.

Promediando estos valores se obtiene una estimación de la altura de la avalancha.

Nuestra curva (figura 5.6) presenta el mismo tipo de comportamiento que el encontrado en la literatura (figura 5.7), pero está notoriamente desplazada hacia ángulos menores. Como nuestras simulaciones no tienen roce, a las partículas les es más fácil fluir, por lo que las avalanchas se detienen en un ángulo mucho menor que en los experimentos.

Una vez obtenido la curva $\theta_{\text{stop}}(h)$ para un conjunto de parámetros, es importante estudiar de que manera afecta en el ángulo de detención modificar los valores de V_{sleep} y V_{WakeUp} .

Dependencia de θ_{stop} en los parámetros V_{sleep} y V_{wakeup}

Se corrieron 8 simulaciones con 2000 partículas normales para un mismo $\dot{\theta} = 10^{-5}$. Como se aprecia en la figura 5.8 el ángulo de detención varía menos de un 2% al variar V un orden de magnitud y no presenta una correlación a la variación de V . Como se observa, algunas de las simulaciones (e.g. $V = 0,10\sqrt{gd}, 0,08\sqrt{gd}$) no terminaron debido a que se produjo colapso inelástico, sin embargo la tendencia del sistema muestra que este efectivamente se

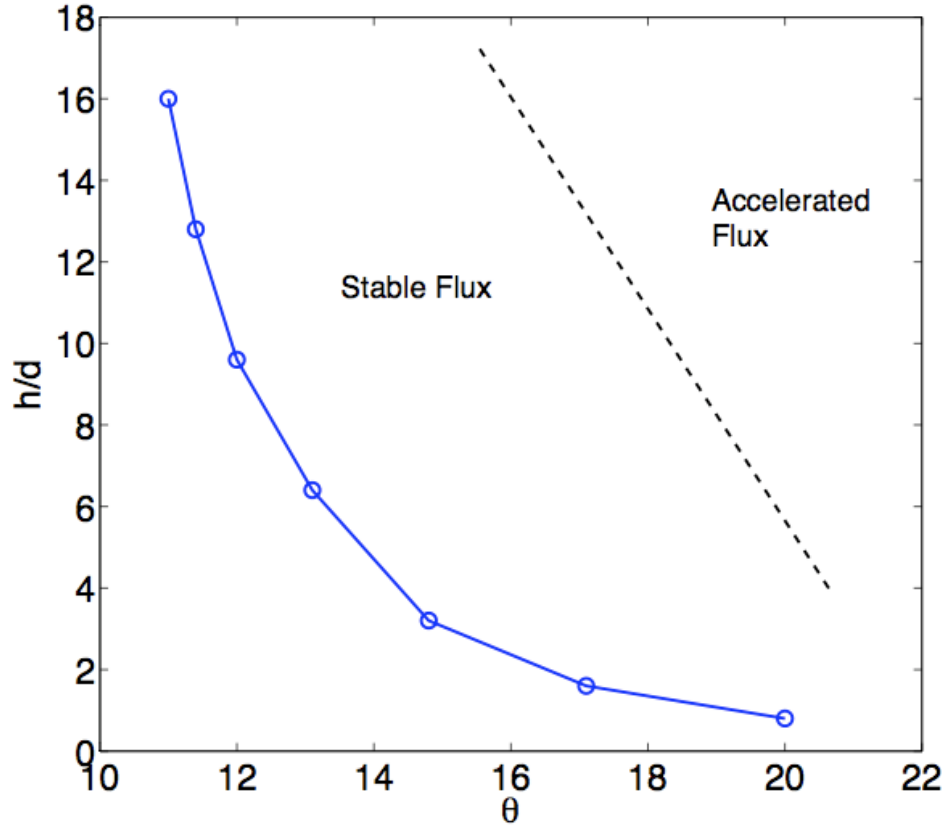


Figura 5.6: Curva $\theta_{stop}(h)$ para nuestro sistema. Encontramos tres regímenes distintos: el sistema no fluye bajo $\theta_{stop}(h)$, fluye a velocidad constante sobre $\theta_{stop}(h)$, y flujos acelerados para ángulos muy grandes. Los errores para la altura son menores que los símbolos, mientras que para determinar el ángulo sólo realizamos una simulación, por lo que es difícil obtener una estadística.

iba a detener. Debido a lo difícil que es seguir estas simulaciones no sabemos la causa de estos colapsos y debemos realizar más estudios.

De esta forma podemos considerar que $\theta_{stop}(h)$ es independiente de los parámetros de nuestro algoritmo para cotas suficientemente chicas ($V \leq 0,1\sqrt{gd}$).

5.4. Proyecciones

Debido a la naturaleza del perfil de velocidad para este tipo de flujos, donde la velocidad es distinta de cero para la capa de partículas en contacto con el suelo, nuestro algoritmo sólo es eficiente al simular la detención del flujo.

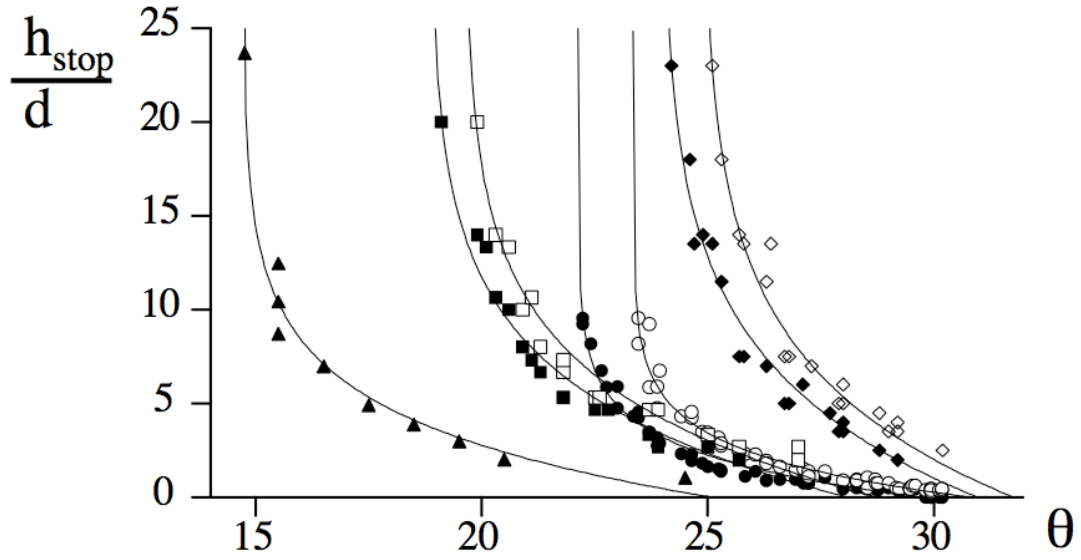


Figura 5.7: Curva $h_{\text{stop}}(\theta)$ para diferentes configuraciones experimentales [35].

Por otro lado, estudiar el ángulo de inicio de la avalancha, partiendo de una configuración estática, es extremadamente eficiente. Puesto que sólo se simulan las partículas despiertas estudiar el comienzo de las avalanchas es algo directo de hacer con nuestro algoritmo.

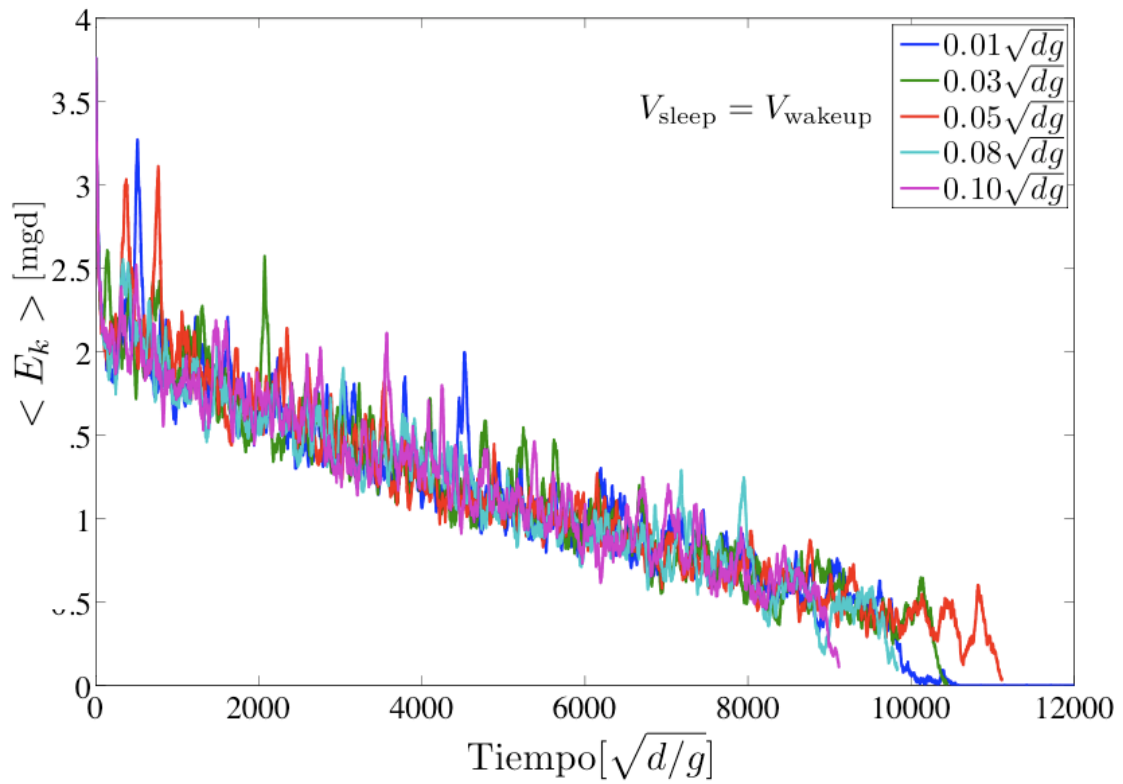


Figura 5.8: Evolución de la energía para una avalancha de 2000 partículas mientras se disminuye el ángulo a una tasa constante para diferentes valores de $V = V_{\text{sleep}} = V_{\text{wakeup}}$.

Capítulo 6

Tambor Rotante

En este capítulo presentamos los resultados obtenidos al estudiar la dinámica del tambor rotante, en particular, los perfiles de velocidad y como afectan nuestras velocidades para dormir y despertar a la dinámica global del sistema. Debido a las discrepancias con los resultados experimentales de las simulaciones preliminares que realizamos para este sistema, tomamos $v_{\text{sleep}} \neq v_{\text{wakeup}}$. Encontramos que nuestras simulaciones presentan comportamientos cualitativamente similares a resultados experimentales y simulaciones anteriores sólo cuando $v_{\text{wakeup}} \gg v_{\text{sleep}}$, es decir, cuando a las partículas les es más difícil despertarse.

6.1. Introducción

El tambor rotante es el caso paradigmático con que comenzamos a pensar el trabajo que concluyó en esta tesis. En él se presenta una fase estática, que se mueve solidaria a las paredes del tambor mientras las condiciones geométricas lo permitan, y para ciertas velocidades angulares se presenta una avalancha estacionaria, donde las partículas se comportan como un fluido granular. Este tipo de dinámica nos obligaba a pensar en que las partículas deben dormir y despertar cíclicamente, algo que no está presente en las simulaciones anteriores donde las partículas despertaban solo por colisiones.

Es importante notar también las implicancias prácticas de poder simular un tambor rotante: en la molienda de cobre se utilizan molinos cilíndricos como el nuestro (ver figura 6.1) para moler el material. Gran parte de la energía inyectada a estos molinos es disipada en choques con las paredes, y tener un modelo que prediga el comportamiento del sistema sería de gran importancia en la optimización de este proceso. En este sentido, poder simular partículas esféricas eficientemente en el tambor rotante es un paso más en el camino de lograr simulaciones realistas de molinos.

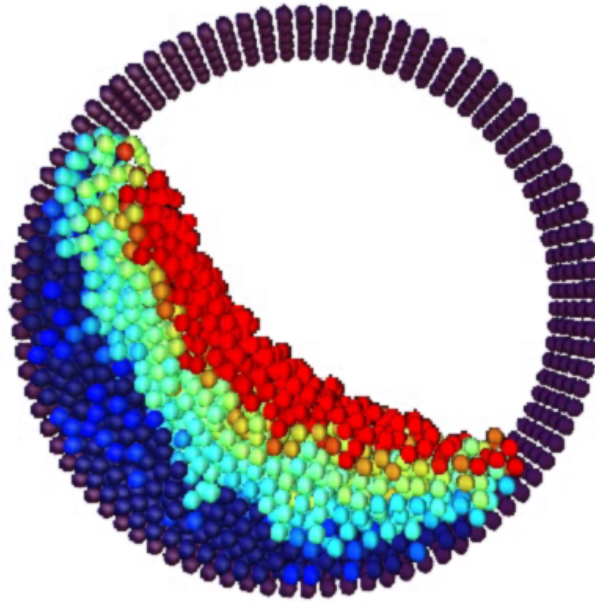


Figura 6.1: Foto para un instante de nuestras simulaciones del tambor rotante con 2000 partículas una vez alcanzado el regimen estacionario. Los colores representan la velocidad de la partícula de azul a rojo en el sistema de referencia rotado, las partículas dormidas estan en azul oscuro, mientras que las que no pueden despertar y forman el borde en morado. El tambor gira en sentido horario.

Por último, en sistemas de este tipo la eficiencia computacional de nuestro algoritmo queda clara, ya que el costo computacional escala como el tamaño de la interfaz entre el sólido y el líquido, es decir, como la superficie del sistema, y no según el volumen del sistema como lo harían la mayoría de los métodos de simulación actuales.

6.2. Sistema

Para construir las paredes del tambor utilizamos partículas sólidas que no se despiertan separadas entre sí por $1,1d$ en la dirección \hat{y} y $1,0d$ de separación de arco en una grilla cuadrada. Es posible generar superficies más complejas que las de un cilindro: como las partículas que no despiertan nunca tienen eventos, ellas pueden estar traslapadas, lo que permite hacer paredes más lisas o bien texturadas.

Para lograr la rotación en un principio pensamos en un nuevo tipo de dinámica de las partículas: la dinámica “tambor”, que considerara un centro de giro y una velocidad angular constante como nuevas variables de las partículas. Con esto, sólo era cuestión de implementar la rutina que encontrara los tiempos de choque entre las partículas normales y las tipo tambor para obtener la rotación. Debido a que las ecuaciones a resolver son complejas preferimos un

enfoque más simple.

Girar la gravedad es equivalente a girar las paredes del tambor en sentido contrario, salvo por el término de la fuerza centrífuga que en nuestro caso es pequeño. Con esto, no debemos agregar ningún estado extra, sino tan sólo cambiar el ángulo de la gravedad a intervalos regulares δt . En las simulaciones que reportamos aquí elegimos $\delta t = 0,2\sqrt{d/g}$.

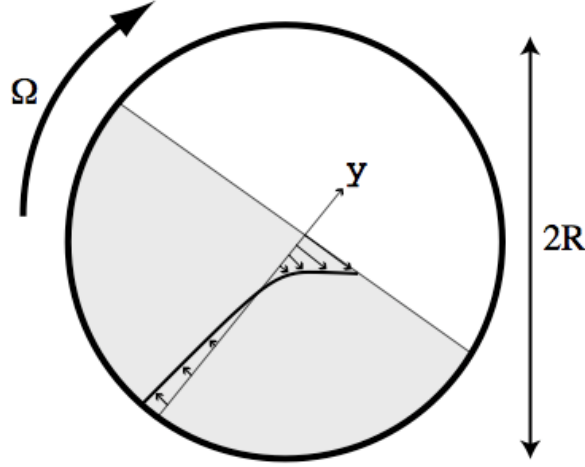


Figura 6.2: Configuración experimental y sistema de coordenadas usado experimentalmente. Imagen tomada de [35].

Para hacer nuestras mediciones usamos un tambor de diametro $D = 50d$ y ancho $w = 4d$. Usamos 6000 partículas para obtener aproximadamente la mitad de llenado, suponiendo que la fracción de volumen de las partículas es $\phi \simeq 0,64$.

6.3. Protocolo

Para la condición inicial distribuimos las partículas de manera uniforme en todo el tambor con velocidad nula y comenzamos la simulación con una velocidad angular fija $\Omega = 0,075/\sqrt{g/d}$, lo que corresponde a un flujo adimensional $Q = \Omega D^2 / (8d\sqrt{gd}) \simeq 23$, durante $100\sqrt{d/g}$ unidades de tiempo, al cabo de las cuales, cada $0,2\sqrt{d/g}$, tomamos 40 fotos del tambor en el sistema de referencia rotado. Esto se repitió para distintos valores de v_{wakeup} .

Para calcular el perfil de velocidades tomamos una franja de grosor d en torno al centro de masa, de manera de emular las mediciones experimentales que se realizan en la zona media del tambor (figura 6.2).

6.4. Resultados

Perfiles de velocidad

Al realizar las simulaciones como veníamos haciendolo hasta el momento con $v_{\text{sleep}} = v_{\text{wakeup}}$, obtuvimos que el perfil de velocidad (curva verde en la figura 6.4) no presenta la cola exponencial característica encontrada en experimentos y simulaciones anteriores para flujos equivalentes como los que se aprecian en la figura 6.3.

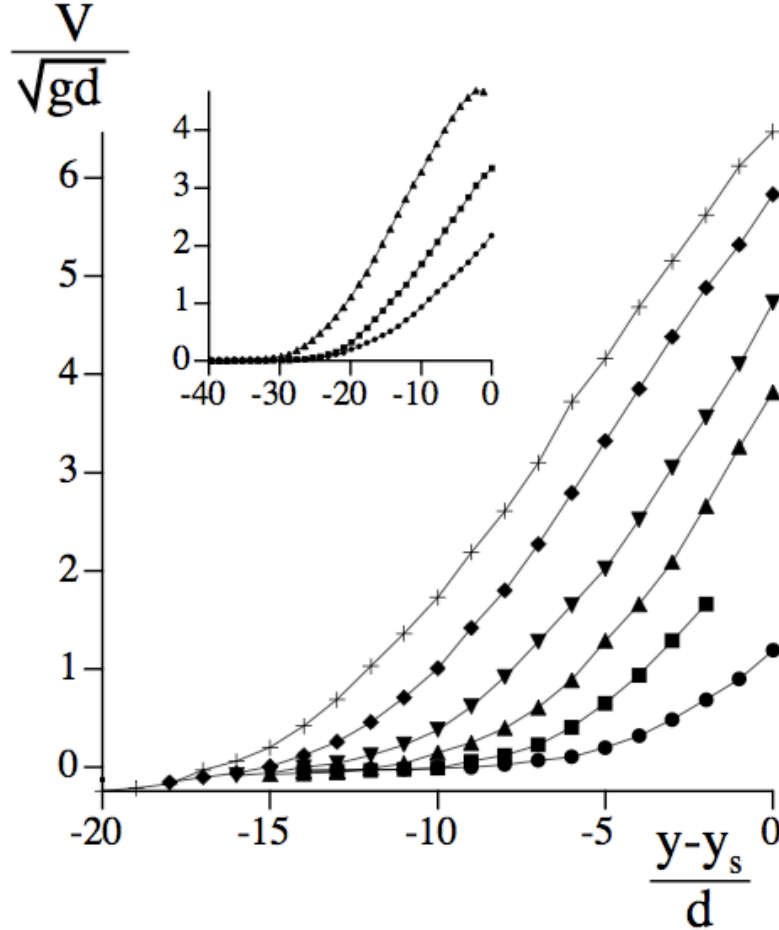


Figura 6.3: Perfil de velocidades experimentalmente obtenidas en [35]. $D/d = 150$, $w/d = 4,66$; del fondo hacia arriba $Q = 5,25; 10,5; 15,75; 21; 30,75; 41,25$. En el inset, perfiles de velocidad para simulación de dinámica de contactos 2D [35]: $D/d = 150$ y $Q = 21; 30,75; 51,75$.

Esta discrepancia nos llevo a ver de qué manera afecta cambiar la velocidad para despertar a las partículas en la forma del perfil de velocidad. Los resultados se observan en la figura 6.4 donde utilizamos $v_{\text{wakeUp}} = 0,01\sqrt{gd}; 0,05\sqrt{gd}; 0,25\sqrt{gd}; 0,50\sqrt{gd}$.

Debido a que nuestras simulaciones no tienen roce, es difícil comparar esta discrepancia

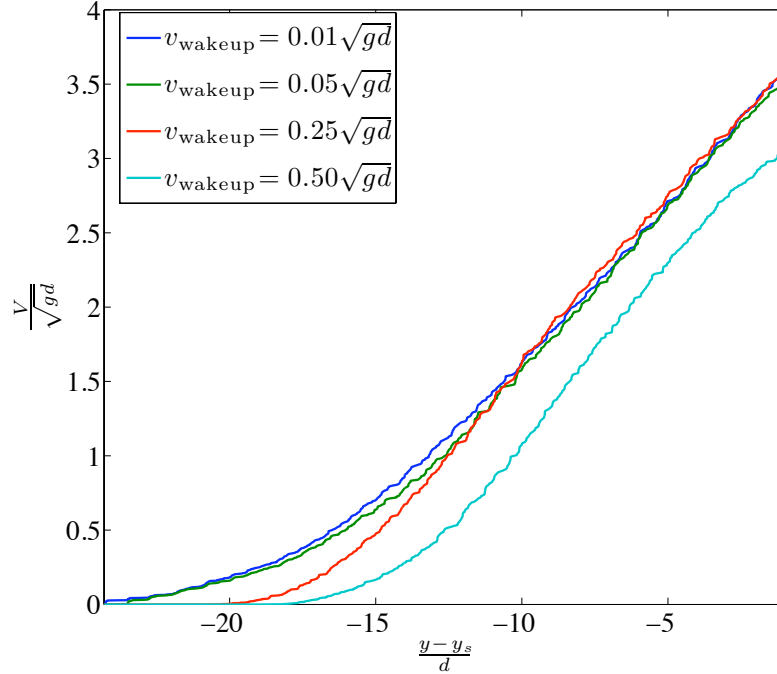


Figura 6.4: Perfil de velocidad para distintas v_{wakeup} .

con los resultados de otros autores [35]: puede significar que el roce es necesario para obtener los perfiles experimentales o bien, y esta explicación es la que nos parece más razonable, que nuestro criterio de despertar es muy permisivo y la energía del sistema es mayor que en sistemas reales.

Esto se explica por lo siguiente: si hay muchas partículas despiertas en el fondo, los choques entre ellas deben ser elásticos, puesto que su velocidad relativa es pequeña. De esta manera, el sistema queda con un remanente de energía que hace que sistema fluya más.

A pesar de esta discrepancia en la zona “sólida”, el flujo lineal en la parte superior del flujo es cualitativamente el mismo para tres de los valores v_{wakeUp} , salvo el caso $v_{wakeUp} = 0,5$ donde la velocidad para despertar es demasiado grande.

Como nuestros valores para dormir y despertar no están determinados a priori y representan una heurística, debemos buscar una forma de calibrar estos valores. Hemos visto que en el caso del tambor rotante las propiedades cualitativas del sistema dependen fuertemente de las cotas del algoritmo, no como en los casos anteriores. De esta manera, el tambor rotante puede ser usado para calibrar los parámetros para otros experimentos.

Debido a que solamente realizamos mediciones para una velocidad angular y con un radio

del tambor para el cual no existen resultados publicados, no es posible ajustar los parámetros en estos momentos. Como trabajo futuro queda realizar comparaciones con simulaciones de dinámica de contactos para ajustar los parámetros.

6.5. Proyecciones

De la manera que actualmente está implementada la rotación del tambor sólo podemos estudiar un rango limitado de velocidades angulares. Es necesario implementar la rotación de las paredes de verdad. Simular partículas no esféricas es algo que nos encantaría hacer pero necesita un estudio mucho más profundo, ya que para hacer aglomerados de partículas la dinámica de los constituyentes se vuelve compleja al igual que las ecuaciones a resolver para encontrar los tiempos de choque. En un futuro cercano queremos utilizar este sistema para calibrar los parámetros de nuestra extensión debido a los cambios cualitativos que se observan en la dinámica al modificar los parámetros, algo que estaba ausente en las configuraciones anteriores.

Capítulo 7

Divertimento

En este capítulo mostramos someramente otros sistemas que son posibles estudiar usando nuestro algoritmo. Primero presentamos el caso del desparrame de partículas en una configuración cuasi-2D y en 2D la que se compara con simulaciones de dinámica de contactos. Luego mostramos el reloj de arena, para el cual sólo se necesitan las condiciones iniciales adecuadas y nuestro algoritmo se encarga de entregar resultados razonables.

7.1. Spreading

Partiendo de una deposición como las realizadas en el capítulo 4, salvo que para un sistema cuasi 2D y más extendido, sacamos las paredes del sistema y lo dejamos evolucionar. Como se aprecia en la figura 7.1, el sistema se desarma por los bordes y se desparrama, llegando a una configuración final con un ángulo de reposo bien definido.

Para comparar con otro método de simulación trabajamos en conjunto con Lydie Staron, Institut Jean le Rond d'Álembert, CNRS-Université Pierre et Marie Curie Paris VI, quien realizó 10 simulaciones de desparrame para sistemas como el de la figura 7.2, esta vez en 2 dimensiones y con discos cuyo diámetro está uniformemente distribuido entre $0,66d$ y $1d$. El método utilizado por ella fue dinámica de contactos [19] mientras nosotros trabajamos con nuestra extensión al método ED.

Para comprar los resultados, medimos la diferencia entre las posiciones finales de las partículas con los dos métodos de simulación. Encontramos que el 78% de las partículas difiere en su posición final en ambas simulaciones menos de $2d$ y que sólo el 4% difiere en más de $10d$: estas son las partículas de los bordes de la avalancha que siguen trayectorias caóticas debido al tipo de suelo que estamos usando. La probabilidad de la diferencia de las posiciones finales sigue una distribución que se presenta en la figura 7.3 donde se ve que la

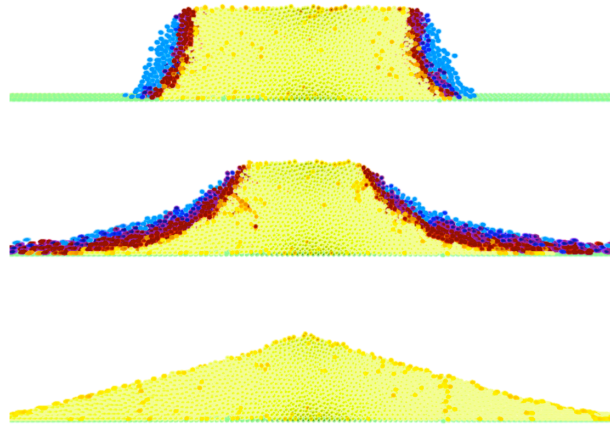


Figura 7.1: Foto para tres instantes de un spreading. Arriba se ve el primer desprendimiento de los bordes; al medio se aprecia la avalancha que se forma; abajo, la configuración final a la que llega el sistema, con ambos ángulos de reposo iguales. El color representa la energía cinética, entre más oscuro, mayor energía

mayor parte de la simulación concuerda.

A fin de disminuir los efectos de los bordes en las comparaciones sería necesario realizar simulaciones con más partículas, pero debido a que en dos dimensiones la disipación es menor que en tres, las simulaciones que intentamos no terminaban debido al colapso inelástico. Una solución puede ser simular partículas más elásticas o modificar los criterios para dormir en dos dimensiones.

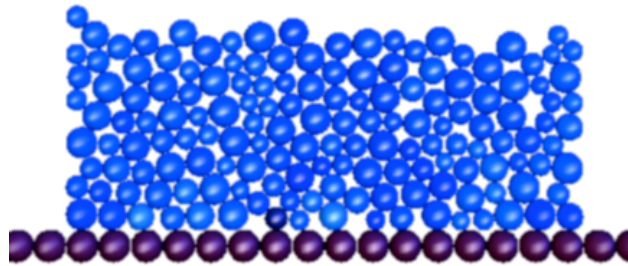


Figura 7.2: Una de las configuraciones iniciales que se estudiaron con nuestro método y con dinámica de contactos.

7.2. Reloj de Arena

Sobre un cono de partículas que no pueden despertar, depositamos 6000 partículas normales y dejamos evolucionar hasta que el sistema queda en reposo. Tomamos la configuración

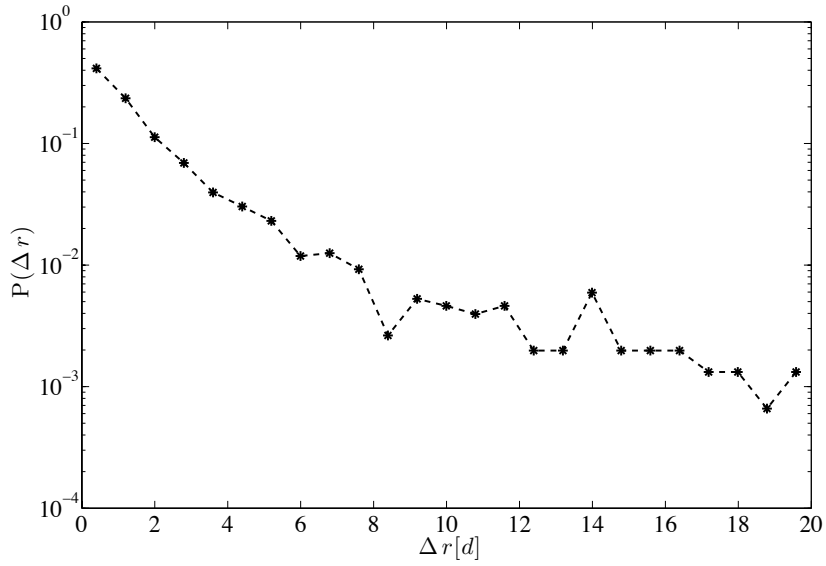


Figura 7.3: Probabilidad para la diferencia de las posiciones finales con ambos métodos.

final y se retiran las partículas que están en el fondo del cono, de manera de dejar una abertura por donde puedan fluir las partículas.

Al perder sus contactos inferiores, las partículas de nuestras simulaciones (figura 7.4) comienzan a fluir por la abertura de una manera cualitativamente similar a la encontrada en experimentos como los de la figura 7.5.

Esta simulación se realizó con los mismos parámetros que la de desparramo. La única diferencia es la condición inicial. Esto nos muestra que nuestro modelo representa la física de sistemas granulares, pues para distintos sistemas, obtenemos el tipo de dinámica esperado sin modificar el modelo de manera ad-hoc.

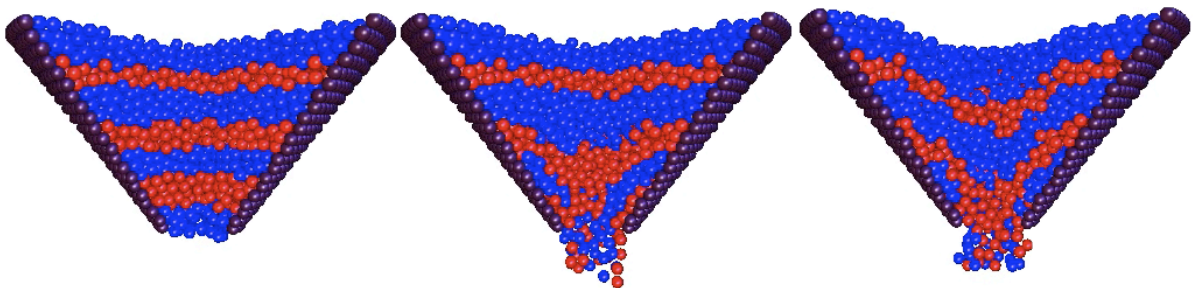


Figura 7.4: Fotos para una sección del reloj de arena en nuestras simulaciones, de izquierda a derecha avanza el tiempo. Las partículas están colorizadas según su altura inicial.

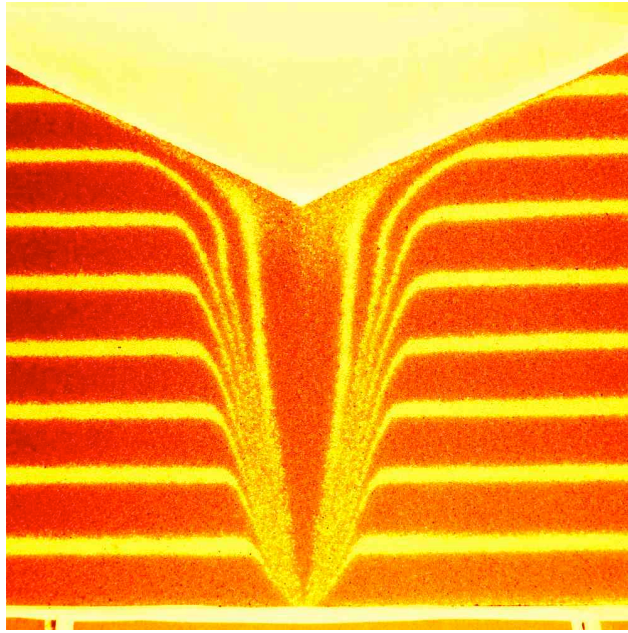


Figura 7.5: Resultados experimentales para flujos a través de un orificio. Foto tomada de [37]

7.3. Proyecciones

Para el desparrame las proyecciones son a lo menos dos: es posible simular sistemas con muchas más partículas que las simulaciones de dinámica de contactos y realizar experimentos tridimensionales en tiempos mucho menores a los métodos actuales.

En el caso del reloj de arena, y su complemento lógico, la pila de arena, se puede estudiar el comportamiento del flujo, nuevamente, con más partículas y en tres dimensiones.

Capítulo 8

Conclusiones

8.1. Algoritmo

En este trabajo hemos presentado una extensión al método ED que permite simular partículas estáticas. Esta simple adición, junto con los criterios para dormir y despertar a las partículas, permite simular de manera eficiente flujos granulares densos en diferentes tipos de superficies.

Gracias a esta extensión hemos logrado simular la existencia de un límite estático en ED. Debido a esto, las simulaciones de flujos granulares densos con regiones estáticas, como el caso del tambor rotante, se vuelven mucho más eficientes.

Al dormir las partículas disminuimos las probabilidades de que ocurra el colapso inelástico. Gracias a esto hemos podido simular sistemas en una región de densidad y disipación nunca antes vista.

Esta extensión incorpora 5 parámetros que tienen que ser ajustados para recuperar la física de los sistemas reales pero que en ciertos rangos no influyen en la dinámica del sistema.

En definitiva, hemos sido capaces de mostrar como sobrellevar dos de los tres problemas del modelo ED. Esto lo hemos logrado al extender la descripción que hacemos de las partículas asignándoles una etiqueta que de cuenta de su estado macroscópico y que consecuentemente modifique la dinámica microscópica de las partículas, para por un lado no perder tiempo computacional en la simulación de choques irrelevantes, y por otro, lograr estados de energía cinética nula como sucede en sistemas reales.

8.2. Deposición

Hemos recuperado la física de las deposiciones, por lo menos en lo que respecta al ordenamiento de la pila resultante. Esto lo medimos con la fracción de empaquetamiento del sistema, donde obtuvimos un resultado acorde con la literatura.

En este experimento nuestro algoritmo es linealmente más rápido que los anteriores ED. Debido a que sólo gastamos tiempo computacional en las partículas que están en la superficie de la deposición, nuestro algoritmo tiene un costo constante en el tiempo, mientras que los anteriores, al simular todas las partículas en la deposición, hace que el costo computacional crezca a medida que más partículas se van agregando. De esta manera, el costo computacional como función del tiempo físico de la simulación es para nuestra extensión $\mathcal{O}(t)$ mientras que para una ED convencional, el costo computacional crece como $\mathcal{O}(t^2)$.

8.3. Avalancha Gruesa

Obtuvimos una fenomenología del ángulo de detención, $\theta_{\text{stop}}(h)$, para avalanchas sobre suelos rugosos, lo que muestra que nuestro modelo simula un material granular efectivamente, a pesar de la ausencia de fricción.

Al realizar este experimento nos dimos cuenta que si bien el ángulo de detención de la avalancha depende fuertemente de la tasa a la cual se cambiaba el ángulo, no hay una correlación entre nuestro parámetro de control, la velocidad para dormir/despertar, y éste. Esto lo interpretamos como que, en la región de valores que exploramos, la física es independiente de nuestras modificaciones algorítmicas.

8.4. Tambor Rotante

Como se aprecia en la figura 6.4, si despertamos las partículas con la misma velocidad con que se duermen, el perfil que obtenemos es cualitativamente distinto a los resultados experimentales y de simulaciones anteriores [35] presentando una zona lineal en el lugar donde debería estar la cola exponencial.

Creemos que la razón de esta discrepancia es que al despertar muy fácilmente las partículas, y debido a que hay que hacer los choques inelásticos cuando la velocidad es pequeña, el sistema queda con un remanente de energía que lo sobrefluidiza, haciendo que el perfil sea cualitativamente distinto.

Esto puede ser comprendido con el siguiente ejemplo: considerense dos partículas detenidas en un plano ligeramente inclinado y rugoso. En un modelo en que no se duermen las partículas, al poner una tercera sobre ellas el sistema empieza a deslizar. En cambio, si la partícula de arriba se duerme y no despierta a sus apoyos en el proceso, el sistema se podría mantener estático.

8.5. Divertimento

Hemos mostrado que nuestro algoritmo es adecuado para la simulación de otros sistemas granulares densos. Por primera vez con ED, es posible estudiar la evolución de sistemas desde y hacia el equilibrio estático, junto con configuraciones donde las partículas cambian de estado constantemente, como el caso del desparrame o el reloj de arena.

Hemos visto que el método ED modificado y dinámica de contactos dan resultados estadísticamente similares al simular desparrames. La ventaja de nuestro algoritmo es que el tiempo de cálculo crece como la cantidad de partículas en la interfaz fluida, mientras que para dinámica de contactos el costo computacional crece con el número de contactos, que es mayor que el número de partículas.

8.6. Trabajo futuro

Al terminar esta tesis hemos mostrado como simular partículas estáticas dentro del marco ED, algo nunca antes hecho. Gracias a que las partículas dormidas no tienen eventos es posible hacer estructuras de diferente rugosidad con ellas, incluso dejandolas traslapadas, a fin de modelar distintos tipos de paredes presentes en sistemas reales. Junto con esto, hemos mostrado criterios para dormir y despertar a las partículas que hacen eficiente la simulación de flujos granulares densos con ED. Estudios preliminares muestran que nuestro algoritmo es a lo menos un orden de magnitud más rápido que dinámica de contactos, que es uno de los métodos usados actualmente en este tipo de simulaciones.

Nuestras partículas no tienen rotación y por lo tanto tampoco tienen roce. La rotación de las partículas en principio no representa mayor problema para nuestra extensión del algoritmo, basta con agregar a las condiciones para dormir que la rotación de la partícula sea nula.

Para simular esferas polidispersas es necesario revisar la condición para despertar, puesto

que ahora es el momentum incidente la cantidad importante. Además, debemos modificar la definición que tenemos para los contactos usados en la revisión de la estabilidad mecánica.

La idea de las distintas dinámicas puede ser extendida en a lo menos dos direcciones. Por un lado, describir de mejor forma a las partículas dormidas, quizá resolviendo una dinámica de contacto para ellas o por una integración de dinámica molecular. Por otro lado, dentro del mismo esquema, se pueden aumentar los tipos de dinámicas distintas en la simulación para simular, por ejemplo, partículas cargadas y neutras en presencia de un campo eléctrico, etc.

Este trabajo comenzó motivado por el caso del tambor rotante y queremos finalizarlo explicando como vemos la simulación del tambor rotante en el futuro. Lo primero que debemos implementar, es el movimiento de las paredes: nuestro enfoque, al cambiar la dirección de la gravedad, no considera el termino centrifugo, que en los molinos usados en la minería es importante. Por otro lado, la diferencia fundamental entre las simulaciones y el fenómeno real es la forma de las partículas: cuando las partículas no son esféricas la dinámica cambia sustantivamente. Nuestra idea es que con el concepto de distintas dinámicas es posible hacer conglomerados de partículas que se muevan como sólido rígido y que puedan romperse. Este enfoque tiene a lo menos dos dificultades técnicas: por un lado, calcular el tiempo de choque entre dos partículas de distintos conglomerados es difícil debido al movimiento que siguen. Por otro lado está el problema de cómo romper estos conglomerados. Ambos problemas merecen de un estudio más detallado pero creemos que se justifica debido a la eficiencia computacional que nuestro método podría otorgar.

Bibliografía

- [1] Rosato, A., Strandburg, K. J., Prinz, F. & Swendsen, R. H. Phys. Rev. Lett. **58**, 1038-1040. (1987)
- [2] Eric Clément, Loic Vanel, Jean Rajchenbach, and Jacques Duran, Phys. Rev. E **53**, 2972 - 2975 (1996)
- [3] J. A. C. Gallas, H. J. Herrmann, and S. Sokolowski, Phys. Rev. Lett. **69**, 1371 - 1374, (1992)
- [4] H. K. Pak and R. P. Behringer, Phys. Rev. Lett. **71**, 1832 - 1835, (1993)
- [5] P. Jop, Y. Forterre, O. Pouliquen, Nature **441**, 727 (2006)
- [6] Daan Frenkel, Berend Smit, Understanding Molecular Simulation: From Algorithms to Applications, Academic Press (2002)
- [7] http://www.math.upatras.gr/~weele/weelerecentresearch_Leidenfrost.htm
- [8] www.vietnamopentour.com.vn
- [9] Nikolai V. Brilliantov and Thorsten Pöschel, Kinetic Theory of Granular Gases. Oxford University Press (2004)
- [10] R. Delannay, M. Louge, P. Richard, N. Taberlet & A. Valance, Nature Materials **6**, 99 - 108 (2007)
- [11] H. Hertz, J. F. reine u. angewandte Math., 92:156, (1882)
- [12] J. Moreau, M. Jean. Numerical treatment of contact and friction: the contact dynamics method. Proc. 3rd Biennial European Joint Conference on Engineering Systems Design and Analysis (1996)
- [13] Francisco Vivanco, Tesis Doctoral, “Flujos en medios granulares secos”, USACH (2006)

- [14] N. V. Brilliantov, F. Spahn, J.M. Hertzsch and T. Pöschel, Phys. Rev. E, 53:5382, (1996)
- [15] D. Weitz, Science, **303**, 5660, 968-969, (2004)
- [16] Cundall P.A., Strack O.D.L., Géotechnique 29, No 1, pp. 47-65. (1979)
- [17] <http://ditwww.epfl.ch/SIC/SA/publications/SCR96/scr8-page4.html>
- [18] Jean, M., Numerical Simulation of geomaterials. pp. 463-486, Salvadurai, A.P.S., Boulon, J.M. (eds.) Elsevier, Amsterdam (1995)
- [19] L. Staron and E. J. Hinch, Granular Matter, **9**, 205-217, (2000)
- [20] D. C. Rapaport, The Art of Molecular Dynamics Simulation, Cambridge University Press (2004)
- [21] B.J. Alder and T.E. Wainwright, J. Chem. Phys. **31** 2, 459-466, (1959)
- [22] M. Marín and P. Cordero, Comp. Phys. Comm. **92**, 214 (1995)
- [23] P. Cordero, M. Marín and D. Risso, Chaos, Solitons and Fractals **6**, 95 (1995).
- [24] D. Risso, R. Soto, S. Godoy and P. Cordero, Phys. Rev. E **72**, 011305 (2005).
- [25] Knight, J. B., C. Fandrich, C. Lau, H. Jaeger and S. Nagel, Phys. Rev. E **51**, 3957 (1995)
- [26] S. Miller and S. Luding, Phys. Rev. E **69**(3), 031305_1-8, (2004)
- [27] S. McNamara and W. R. Young, Phys. Rev. E **50**, R28 (1994).
- [28] M. G. Clerc, P. Cordero, J. Dunstan, K. Huff, N. Mujica, D. Risso & G. Varas, Nature Physics 4, 249 - 254 (2008)
- [29] Nikolai V. Brilliantov and Thorsten Pöschel, Phys. Rev. E **61**, 1716 - 1721 (2000)
- [30] Rosa Ramírez, T. Pöschel, N. V. Brilliantov, and Thomas Schwager, Phys. Rev. E 60 4465-4472 (1999)
- [31] S. Luding, and S. McNamara, Granular Matter **1(3)**, 113-128, (1998)
- [32] M. Marín, D. Risso, and P. Cordero, J. Comput. Phys. **109**, 306 (1993).
- [33] S. Torquato, T. M. Truskett, and P. G. Debenedetti, Phys. Rev. Lett. **84**, 2064 (2000).

- [34] H. Hertz, J. für die reine u. angew. Math., **92**, 136 (1882).
- [35] GDR MiDi, Eur. Phys. J. E **14**, 341 (2004).
- [36] <http://www.geometrictools.com/Documentation/PolynomialSystems.pdf>
- [37] <http://physics.clarku.edu/~akudrolli/silo.html>