



Universidad Central "Marta Abreu" de Las Villas
Facultad de Matemática-Física-Computación

Año 2006

Método de Elementos Distintos

Autor:
Irvin Pablo Pérez Morales

Tutores:
Dra. Lucía Argüelles Cortés
Dr. Ing. Carlos A. Recarey Morfa

Agradecimientos.

Deseo agradecer a mis tutores Lucía y Recarey, a mis dos abuelas, a mi madre y a mi padre, a mi hermano el Muķe, a mis hermanos Raian y Kenier, a Patricia, a mis amigos Yordanis y Roberto, a mi familia y a todas las personas que han contribuido con la realización de este trabajo, y a Amarilys la secretaria, por sacarnos del aprieto.

Resumen

Se ha llevado a cabo una búsqueda bibliográfica con el objetivo de conocer el estado del arte de los métodos de generación de un medio para realizar simulaciones a través del Método de Elementos Distintos y para estar al tanto de las formulaciones y aplicaciones de este último método, después de lo cual se han expuesto las conclusiones parciales que justifican la realización de la presente investigación.

Se ha formulado un algoritmo de generación del medio que no depende del tipo de partícula con la que se haga el empaquetamiento y se ha formulado detalladamente cómo dicho algoritmo puede ser usado con partículas de gran variedad de formas. También se ha formulado un algoritmo que permite determinar si un punto se encuentra dentro de un conjunto de segmentos(caso 2D) o dentro de una malla(caso 3D). Este algoritmo puede ser empleado dentro del mencionado anteriormente. Este algoritmo intenta subsanar desventajas que hacen fallar en algunos casos a otros algoritmos similares. Además, se han presentado ejemplos de medios generados con el mencionado algoritmo de generación del medio.

Se ha hecho una formulación detallada del Método de Elementos Distintos para partículas esféricas y formadas por colisiones de esferas, y más adelante se ha presentado un ejemplo de la aplicación de este método a un problema de ingeniería.

Por último, se ha arribado a las conclusiones de la investigación, exponiéndose a la vez las direcciones hacia las cuales será extendida la presente tesis.

Summary

There has been made a search of bibliography with the aim of knowing the state of art of the methods of generation of a packing for carrying out simulations with the Discrete Element Method, and in order to be aware of the formulations and applications of this method, after which have been presented the partial conclusions that justify the present research.

There has been formulated an algorithm of packing which does not depend on the type of particle to make the packing and there has been formulated in detail how this algorithm can be used with particles of several shapes. There has also been formulated an algorithm which allows to determine whether a point lies within a set of segments (case 2D) or within a mesh (case 3D). This algorithm can be employed as a part of the previously mentioned and does not have some disadvantages of similar algorithms. Besides, there have been presented examples of packings generated with this algorithm of generation.

There has been made a detailed formulation of the Discrete Element Method for spherical particles and for particles formed by the collision of spheres. There has also been presented an example of the application of this method to solving an engineering problem.

Finally, there have been stated the conclusions of the research, saying at the same time the future expansions this work will have.

Índice

	Pags.
Introducción.	1
Capítulo 1. Estado del arte.	10
1.0 Estado del arte de la modelación de medios discontinuos.	10
1.1 Introducción	10
1.2 Generalidades de la simulación numérica de medios discontinuos.	11
1.3 Generación del medio para modelación de problemas físicomatemáticos con el método de las partículas o método de elementos discretos.	12
1.4 Formulación y método de elementos de partículas.	19
1.4.1 Método de partículas con enfoque macroestructural.	19
1.4.2 Método de partículas con enfoque microestructural.	26
1.5 Conclusiones parciales.	30
Capítulo 2. Generación del medio.	33
2.0 Introducción.	33
2.1 Detección de contactos.	33
2.2 Construcción de una circunferencia tangente a otras dos.	33
2.3 Construcción de una esfera tangente a otras tres.	34
2.4 Construcción de una elipse tangente a otras dos.	35
2.5 Construcción de un elipsoide tangente a otros tres.	37
2.6 Aproximación de elipses y elipsoides de revolución mediante circunferencias y esferas respectivamente.	40
2.7 Construcción de una partícula formada por colisión de circunferencias tangente a otras dos(caso 2D).	41
2.8 Construcción de una partícula formada por colisión de esferas tangente a otras tres(caso 3D).	43
2.9 Construcción de un polígono que está en contacto con otros dos.	44
2.10 Construcción de un poliedro que está en contacto con otros tres.	46
2.11 Método de interioridad en dos dimensiones.	48
2.12 Método de interioridad en tres dimensiones.	50
2.13 Algoritmo de empaquetamiento de partículas (en tres dimensiones).	52
2.14 Aplicaciones preliminares en la generación de diferentes geometrías.	54
2.15 Conclusiones parciales.	57
Capítulo 3. El método de elementos discretos.	58
1.0 Introducción y aspectos generales de la formulación del método de los elementos discretos.	58
2.0 Proceso de cálculo del método de los elementos discretos.	61

3.0	Formulación del método de los elementos discretos.	63
3.1	Ecuaciones de Movimiento.	63
3.2	Amortiguamiento global.	64
3.3	Determinación de la densidad y masa micro-estructural.	66
3.4	Integración del problema con un esquema de diferencia finita central.	67
3.5	Estabilidad numérica del esquema de integración explícito.	69
3.6	Especificidades de la lógica de cluster de esferas o conglomerados.	71
3.6.1	Propiedades de masa de un conglomerado.	72
3.6.2	Ecuaciones completas de movimiento para un conglomerado.	72
3.6.3	Ecuaciones aproximadas para el movimiento de un conglomerado.	77
3.7	Modelo constitutivo de contacto micro estructural y evaluación de las fuerzas de contacto.	77
3.8	Búsqueda y Chequeo de Contacto.	86
3.8.1	Generalidades al problema de Contacto.	86
3.8.2	Detección de contacto.	87
3.8.3	Problema de Contacto entre elementos discretos esféricos.	88
3.8.3.1	Subdivisión en celdas.	89
3.8.3.2	Árboles binarios.	89
3.8.3.3	Celdas centradas en el cuerpo.	90
3.8.3.4	Spatial heap sort y Spatial sorting.	91
3.8.3.5	NBS.	91
3.8.3.5.1	Generalidades.	91
3.8.3.5.2	Definición del Problema de Detección de Contacto en el algoritmo NBS.	92
3.8.3.5.3	Descomposición del Espacio en el Algoritmo NBS.	92
3.8.3.5.4	Detección de Contacto en el Algoritmo NBS.	94
3.8.3.6	Quadtree (2D) y Octree (3D).	94
3.8.3.6.1	Búsqueda global.	95
3.8.3.6.2	Búsqueda local.	96
3.9	Aplicación preliminar del método de partículas.	96
3.10	Análisis de los resultados de la simulación numérica preliminar.	98
3.11	Conclusiones parciales.	99
Conclusiones		100
Recomendaciones		102
Referencias Bibliográficas		103
Bibliografía		110
Apéndices.		
Apéndice I: Método de Montecarlo.		
Apéndice II: Implementaciones.		

Introducción.

El hombre, a lo largo de los años ha adoptado una posición científica al tratar de estudiar los fenómenos físico-naturales que ocurren a su alrededor. En este proceso ha aceptado un aumento de responsabilidad y esfuerzo, problema éste que condiciona incluso el desarrollo de la humanidad. En este camino ha podido elegir entre dos grupos de soluciones: ahorrar sus esfuerzos y limitarse a tomar lo que la naturaleza le brinda casi directamente o intentar arrancarle el máximo de ventajas y satisfacciones materiales a cambio de un conocimiento y dominio previo de las leyes del mundo material.

Con este fin, en los últimos tiempos ha estado en boga el desarrollo y empleo de las técnicas de modelación para intentar investigar estas leyes y fenómenos. En este caso se tiene especial interés en el desarrollo de nuevos métodos numéricos y de generación del medio para resolver complejos problemas de la geometría computacional, mecánica computacional y la ingeniería. Al respecto, la modelación y la matemática numérica han adquirido un carácter de métodos científicos generales que, en esencia, penetran todas las esferas de la actividad cognoscitiva y transformadora del hombre, con especial énfasis en la Ingeniería Aplicada. Enriquecido por las conquistas y desarrollo de la Matemática, la Cibernética y el enfoque sistémico, la modelación posibilita la profundización de los conocimientos sobre el mundo circundante y se convierte en medio de dirección y de toma de decisiones racionales sobre problemas de utilización de la naturaleza.

El aumento de la importancia del método de la modelación en el conocimiento científico está determinado, ante todo, por la lógica interna del desarrollo de la ciencia; en particular, por la frecuente necesidad de un reflejo mediatizado de la realidad objetiva. En el desarrollo y proliferación de la modelación y los métodos numéricos han tenido una gran trascendencia las consideraciones económicas relacionadas con el aumento de la efectividad de las investigaciones científicas y la optimización de la actividad humana, en términos generales.

Debe señalarse que, a pesar de que en los últimos años haya aumentado la intensidad de las investigaciones en el campo de la matemática numérica, la modelación, la problemática metodológica de este importante método del conocimiento científico moderno está muy lejos de haber sido agotada. Importantísimos problemas gnoseológicos de la modelación deben ser estudiados minuciosamente a la luz de las circunstancias de la ciencia de nuestros días. Esta situación puede explicarse por la circunstancia de que el método de la modelación en la ciencia actual es muy complicado y diverso y, lo que es fundamental, se encuentra en un estado de permanente enriquecimiento y desarrollo. La mecánica computacional y la geometría computacional aplicada a la resolución de complejos problemas de la ingeniería, no han quedado ajenas a este desarrollo, por lo que se buscan nuevos enfoques con carácter general y sistémico, para la solución de los diferentes problemas que enfrentan los ingenieros. De lo anteriormente señalado se desprende la importancia de la matemática, la geometría computacional y la mecánica computacional en la resolución de los problemas complejos de la física matemática y la ingeniería.

Estos aspectos han propiciado el desarrollo de nuevos métodos numéricos y de generación del medio, con el empleo de los novedosos modelos constitutivos que describen el comportamiento de los diferentes materiales que conforman la edificación, estructura u obra de ingeniería y por qué no, el terreno.

A lo expresado con anterioridad se le añade que el objetivo central del mundo en general para los próximos tiempos, tiene que ser la eficiencia, pues constituye una de las mejores potencialidades de que se dispone. Por ende, hacer un mejor uso de los recursos y alcanzar mejores resultados con menos costo tiene que ser la estrategia económica número uno, la que debe estar apoyada por una correcta política científico-técnica.

Desde este punto de vista, vienen los métodos numéricos, la geometría computacional y la modelación a desempeñar un papel decisivo como piedra angular del proceso de análisis y diseño de problemas de ingeniería. La mejor forma de lograr lo anterior, es abordar las soluciones de los complejos problemas ingenieriles de la mejor manera posible, con el empleo de las nuevas técnicas numéricas y de modelación sistémica, las cuales se sustentan, de forma indirecta, en criterios racionales y eficientes. En este sentido, la búsqueda de nuevas técnicas de generación del medio, numéricas, y de modelación que incluyan la naturaleza microestructural de los fenómenos, posibilitan obtener solicitaciones (esfuerzos mecánicos y desplazamientos) más reales, en el proceso de análisis de estructuras, lo que se traduce en soluciones más eficientes y económicas en el proceso de diseño de las diferentes obras de ingeniería.

El desarrollo alcanzado hasta la actualidad en la temática de geometría computacional, mecánica computacional, métodos numéricos y modelación de los problemas de ingeniería ha quedado ajeno a la esencia microestructural de estos fenómenos físicos relacionados con estas disciplinas. Solo existen someros intentos, que no se basan en un enfoque sistémico del proceso de modelación del estado tenso-deformacional desde el punto de vista microestructural y su extrapolación a un medio discreto.

Se buscan nuevas técnicas de generación del medio, numéricas y de modelación que se basen en un enfoque sistémico que incluyan la naturaleza microestructural de los fenómenos físicos. Precisamente, esta es la idea básica del contexto investigativo en que se enmarca la presente tesis, con el afán de resolver lo concerniente a la modelación físico-matemática del estado tenso-deformacional de un material en un punto desde el punto de vista microestructural y su extrapolación al medio discontinuo, para solucionar la modelación y análisis de diversos problemas de ingeniería. En el proceso de resolución de esta problemática se hace necesario el desarrollo de nuevos métodos numéricos que tomen en cuenta la naturaleza microestructural del medio y además de trabajar en formulaciones matemáticas relacionadas con la geometría computacional para establecer métodos de generación del medio que posibiliten el empaquetamiento de partículas.

El estudio de modelos de partículas a través del Método de los Elementos Distintos con el empleo de nuevos modelos constitutivos micro-estructurales es un tema de indudable interés científico en el campo de los métodos numéricos en la ingeniería y es un aspecto que guarda singular significación con lo ya expresado con anterioridad.

El desarrollo de la mecánica computacional siempre ha ido paralelo al intento de aproximar mediante modelaciones y simulaciones numéricas la esencia de los fenómenos físicos. Los modelos discretos (modelos de partículas) se basan en el estudio de los problemas de ingeniería a una escala cada vez menor, en la actualidad, hasta la escala microscópica. Todo esfuerzo encaminado al acercamiento del micro-mundo y al estudio de sus leyes y comportamiento es una tarea de constante investigación y es precisamente en esta tendencia de trabajo donde se centra la finalidad de esta investigación.

La finalidad de acercarse cada vez más al micro-mundo y a la esencia micro-estructural de los fenómenos físicos, permite realizar estudios de seguridad estructural con un alto grado de precisión y fiabilidad, lo que a su vez posibilita prever fallos estructurales severos, ya que los modelos discretos (modelos de partículas) permiten detectar fallos a nivel micro. El hecho de poder estudiar los estados tensionales y deformacionales a un nivel microscópico posibilita detectar la aparición de microfisuras y discontinuidades que son el comienzo de las cadenas de fallos estructurales y la formación de micro-zonas de plastificación que culminan desafortunadamente a nivel macroscópico en el fallo estructural. Estas delimitaciones no apreciables por los métodos macroscópicos convencionales (Método de elementos finitos - MEF-, Método de elementos de Contorno - MEC -, Método de Diferencias Finitas -MDF-, Método de elementos Finitos de Puntos -MEFP-), posibilitan estudiar la evolución de las micro-grietas y permiten prevenir catástrofes y desastres que pueden afectar a las estructuras y el medio ambiente. Esta es la razón fundamental para desarrollar métodos numéricos que posibiliten realizar estudios a nivel microscópico y es por eso el objetivo primordial del presente trabajo que hoy por hoy es un tema de singular interés e importancia para los centros de investigación de élite mundial.

El desarrollo de nuevas formulaciones que posibiliten la generación del medio a través de técnicas de empaquetamiento de partículas, la evolución del método de los elementos distintos o método de partículas y la implementación de nuevos modelos constitutivos (micro-estructurales) constituye un elemento importante dentro del desarrollo de la mecánica computacional. La solución de diversos problemas que propician los fallos estructurales producto a catástrofes naturales (sismos, huracanes, deslizamientos de taludes, impacto de explosivos, etc); los cuales afectan severamente a la naturaleza, la biosfera, el medio ambiente, el ecosistema y a la vida social de la humanidad son temas de investigación de relevancia internacional. El estudio de estos fenómenos a través de técnicas de simulación, dirigiendo el interés de las investigaciones a la esencia más micro-estructural de los fenómenos naturales propiciará tener un conocimiento más detallado de fenómenos y su incidencia en las estructuras, posibilitando prevenir y mitigar con mayor certeza, seguridad y fiabilidad estos fenómenos.

Fundamentos metodológicos del trabajo

Situación Problemática:

La búsqueda de métodos de generación del medio (empaquetamiento de partículas) simulación y modelación que se acerquen cada día más a la realidad de los fenómenos físicos naturales es la esencia de la ciencia moderna. Es por ello que el desarrollo de métodos de modelación que trabajen a escala de la micromecánica es un aspecto esencial dentro del desarrollo de la Mecánica Computacional y los Métodos Numéricos en la Ingeniería. La inexistencia de métodos que

describan los mecanismos de fallos, desde un nivel micro con la formación de microfisuras, hasta fisuras y grietas a nivel macro estructural es un aspecto que es tema de investigación de los diferentes centros de investigación de prestigio internacional. A lo anteriormente expresado se le une las diferentes políticas internacionales relacionadas con la preservación del medio ambiente y el ahorro de los recursos materiales disponibles en la naturaleza o creados por el hombre. Estos aspectos presuponen que las ciencias básicas y en especial la matemática este en función de resolver complejos problemas basándose en el desarrollo de las técnicas numéricas y de geometría computacional.

Problemas:

¿Qué métodos de generación del medio o de empaquetamiento de partículas se pueden formular para que sirva de preámbulo para la aplicación del método de partículas?

¿Qué métodos numéricos de modelación y simulación de problemas de ingeniería que trabajen a nivel de la escala de la micro-mecánica se pueden formular para dar respuesta a los fenómenos físicos que se desarrollan a nivel micro y macro respectivamente?

Interrogantes

¿Será posible establecer un método de generación del medio que posibilite empaquetar un sistema de partículas?

¿Se podrá establecer un método de modelación y simulación de problemas de ingeniería que trabajen a nivel de escala de la micro-mecánica?

Objeto de estudio

En correspondencia con lo planteado con anterioridad, queda definido el planteamiento del problema, es decir, *el objeto de estudio* de la presente investigación consistente en el desarrollo de métodos de generación del medio para empaquetar partículas y continuar contribuyendo a la evolución del método de partículas o de elementos distintos. En este caso el enfoque de la investigación está dirigido al desarrollo de métodos numéricos capaces de estudiar los fenómenos micro-mecánicos donde se delimita la microestructura de los materiales y los microdefectos que en una concatenación de fallos se transforman en fisuras, grietas y colapsos de las edificaciones y el terreno.

Conocido el objeto de estudio de la investigación es necesario definir la hipótesis de trabajo de la misma:

La **hipótesis** de la presente investigación es, en esencia, que el desarrollo de métodos de generación del medio para el empaquetamiento de partículas y la contribución en la evolución del método de partículas para resolver diversos problemas de la mecánica computacional, posibilita una mejor correspondencia entre el modelo y el fenómeno físico real, lo que implica una mejor modelización de los problemas ingenieriles, permitiendo, a su vez, obtener soluciones más fiables, acabadas y racionales.

Delimitados el objeto de estudio y la hipótesis de la investigación se procede a definir los objetivos (generales y específicos) y las tareas científicas:

Objetivo general:

Desarrollar una formulación que posibilite la generación del medio a través del empaquetamiento de partículas y contribuir a la evolución y desarrollo del método de partículas con la finalidad de poder realizar un estudio sistémico del comportamiento físico-matemático del estado tenso-deformacional de un material desde un punto de vista microestructural y su extrapolación para la modelación del medio discontinuo (o continuo) para dar solución a distintos problemas ingenieriles.

Objetivos específicos:

- Formulación de un método de generación del medio a través del empaquetamiento de partículas.
- Formulación del Método de los Elementos Distintos (MED) que posibilite realizar estudios a escala de la micromecánica donde se delimite la microestructura de los materiales y los microdefectos que en una concatenación de fallos se transforman en fisuras, grietas y colapsos de las edificaciones y el terreno.

Tareas científicas a acometer:

- Recopilación bibliográfica preliminar, definición, aprobación del tema y elaboración del plan de trabajo.
- Estudio bibliográfico y análisis del estado del arte de la temática.
- Redacción de la primera versión del Capítulo I acerca del estado del arte de la generación del medio a través de técnicas de empaquetamiento de partículas y modelación con método de partículas.
- Formulación del método de generación del medio a través de técnicas de empaquetamiento de partículas.
- Redacción de la primera versión del Capítulo II “Generación del medio”.
- Formulación del Método de los Elementos Distintos (MED).
- Redacción de la Primera versión del Capítulo III “Método de elementos Discretos”.
- Redacción de la primera versión de las “Conclusiones y Recomendaciones” del trabajo.
- Análisis del contexto global de la tesis y redacción definitiva de la misma.

Novedad científica:

Desarrollo de una formulación que permite realizar la generación de un medio a través de técnicas de empaquetamiento de partículas que posibilitan estudiar diferentes problemas de la mecánica computacional a través del método de partículas.

Aportes científicos relevantes:

- El establecimiento de una formulación del Método de los Elementos Distintos (MED) donde se puedan emplear diversos tipos de partícula, aspecto no abordado en las investigaciones precedentes.

Valor Científico de la Investigación.

El valor científico de la investigación se centra en el hecho de que con las formulaciones desarrolladas se pueden estudiar los estados tensionales y deformacionales a un nivel microscópico, posibilitando detectar la aparición de microfisuras y discontinuidades que son el comienzo de las cadenas de fallos estructurales y la formación de micro-zonas de plastificación que culminan desafortunadamente a nivel macroscópico en el fallo estructural. Este tipo de estudio no se puede realizar con los métodos clásicos convencionales, aspecto que delimita el valor científico de la investigación.

Valor Metodológico de la Investigación.

El valor metodológico de la investigación se evidencia a través de dos aspectos fundamentales:

- El desarrollo de una formulación de generación del medio con un carácter general que posibilita empaquetar por el mismo procedimiento diversos tipos de partículas, aspecto que puede tomarse de base para desarrollos futuros.
- Contribución a la evolución del método de elementos de partículas donde se establecen las peculiaridades para emplear diversos tipos, aspecto que sienta la bases para nuevas geometrías.

Valor Práctico Ingenieril de la Investigación.

El valor práctico de esta investigación está fundamentado sobre la base de que se dispone de dos formulaciones: la primera permite la generación del medio con la finalidad de empaquetar partículas, y la segunda es una formulación del método de partículas, que permite detectar la aparición de microfisuras y discontinuidades que son el comienzo de las cadenas de fallos estructurales y la formación de micro-zonas de plastificación que culminan desafortunadamente a nivel macroscópico en el fallo estructural. Desde el punto de vista práctico ingenieril este aspecto es de singular importancia ya que el método permite estudiar la mecánica de fallos geotécnicos y estructurales a escala de la micro-mecánica y a su vez posibilita resolver, bajo estas condiciones, diversos problemas de ingeniería.

Beneficios de la investigación.

Las formulaciones desarrolladas en la investigación y su implementación computacional servirán para generar y empaquetar partículas de formas diversas con la finalidad de modelizar y simular numéricamente (modelo de partículas) diversos problemas de la física matemática y la mecánica computacional con la peculiaridad y novedad científica que en este caso se podrá investigar los fenómenos hasta un nivel microscópico u otra escala de la mecánica computacional superior a esta. Se puede aplicar a aquellos problemas de ingeniería donde el riguroso conocimiento y estudio micro estructural y microscópico constituya un aspecto importante en su seguridad estructural. La simulación numérica de voladuras, estabilidad de laderas, minas a cielo abierto,

etc. , donde su investigación tenga una marcada influencia en la preservación del medio ambiente, son problemas a abordar y estudiar, con la singularidad de que el estudio se realiza a un nivel elemental de las micro-partículas que conforman el medio. La simulación de problemas convencionales de ingeniería y de mecánica computacional es otro de los problemas que no escapan a esta formulación.

Metodología general de investigación.

Para dar cumplimiento a los objetivos trazados en la tesis y tomando en cuenta el volumen de tareas científicas a acometer fue necesario organizar y estructurar el trabajo en varias etapas, las cuales definen la metodología general de investigación.

La metodología de investigación concebida incluyó:

Definición del problema de estudio:

Ya en los fundamentos conceptuales y metodológicos de la tesis se explicó la necesidad de la investigación en cuanto a la generación del medio a través de empaquetamiento de partículas y otro aspecto abordado es la modelación de los fenómenos a escala de la micro-mecánica.

Recopilación bibliográfica:

El estudio de la bibliografía se efectuará en distintas etapas en correspondencia con las exigencias que va generando el trabajo, por lo que aunque se ubicó al comienzo de la investigación, el mismo se realizara de forma sistemática hasta el final del trabajo. Nacionalmente se efectuaron búsquedas bibliográficas automatizadas en los principales centros de información, así como por los métodos clásicos. La búsqueda bibliográfica incluye el empleo de sistemas de búsqueda de avanzada (current content, Internet, etc.).

Formación de la base teórica: Por la complejidad matemática y estadística del tema tratado en la tesis se hace necesario tener una formación adicional en estas disciplinas, que permite enfrentar las distintas fases del desarrollo investigativo del trabajo; realizándose la formación de la base teórica relacionada con la tesis, específicamente, en temas de geometría computacional, en específico la generación de un medio a través de empaquetamiento de partículas, y el desarrollo de la mecánica computacional , contribuyendo a la evolución del método de partículas, aspecto que posibilita la modelación físico-matemática del comportamiento de materiales y de la mecánica del cuerpo deformable.

Diseño general de la investigación:

En esta etapa se efectuó la concepción general de la investigación, se definieron las hipótesis, los objetivos, tareas científicas y la estrategia de trabajo para enfrentar la investigación. Estos aspectos fueron abordados con anterioridad y con los mismos queda definida la metodología de investigación. Seguidamente, se estructuró la investigación en fases para ir dándole respuesta a cada uno de los objetivos y tareas científicas propuestas.

Desarrollo particular de cada fase de la investigación:

El desarrollo de la investigación queda estructurado en tres fases. La primera relacionada con el estudio bibliográfico y análisis del estado del arte de la temática, lo que posibilita justificar el desarrollo de la investigación. En esta fase se exponen los antecedentes y el estado actual de la temática, destacándose los fundamentos teóricos principales.

La segunda etapa esta relacionada con la formulación de un método de generación del medio a través de técnicas de empaquetamiento de partículas. En este caso logra formular un algoritmo que permite generar diversos tipos de geometrías con disímiles partículas, siendo el mismo independiente de la geometría de las partículas a emplear.

La tercera se relaciona con la formulación de MED, el cual posibilita enfrentar problemas de la física matemática desde la escala de la mecánica estructural hasta la micromecánica. Con esta formulación se hace una contribución marcada al campo de la ciencia a lográndose extender el método con la posibilidad de emplear varios tipos de partícula.

Elaboración de conclusiones y recomendaciones:

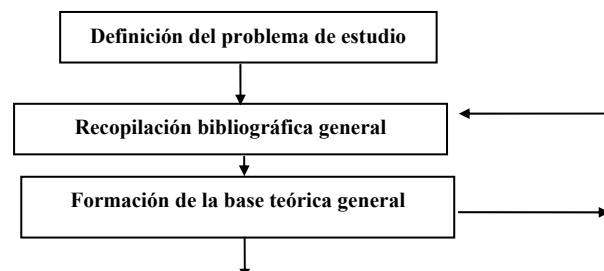
Finalmente se procederá al análisis integral de los resultados obtenidos en cada una de las fases del desarrollo de la investigación, formulándose en cada caso las correspondientes conclusiones parciales, las generales, así como las recomendaciones definitivas del trabajo.

Estructura del trabajo:

La estructura de la tesis guarda una relación directa con la metodología de investigación establecida y, específicamente, con el desarrollo particular de cada una de las fases de la investigación. La misma se encuentra formada por una introducción general, tres capítulos, las conclusiones, recomendaciones y bibliografía, así como los apéndices necesarios.

El orden y estructura lógica del trabajo se establece a continuación:

- ***Resumen***
- ***Introducción***
- ***Capítulo I: Estado del arte.***
- ***Capítulo II: Generación del medio.***
- ***Capítulo III: El método de elementos discretos.***
- ***Conclusiones***
- ***Recomendaciones***
- ***Bibliografía y Referencias Bibliográficas.***
- ***Apéndices***



Campo de aplicación:

El campo de aplicación de este trabajo está estrechamente vinculado con el proceso de modelización y análisis de los problemas de la física matemática y la mecánica computacional. En este sentido, la búsqueda de nuevas técnicas de modelación que posibiliten modelar problemas de la física matemática, desde la escala de la mecánica estructural hasta la micromecánica, posibilitan obtener solicitaciones (esfuerzos mecánicos y desplazamientos) más reales, en el proceso de estudio, lo que se traduce en soluciones más eficientes y económicas en el proceso de diseño de ingeniería.

Aprobación del trabajo:

- Presentación y premio en el Forum de Estudiantes de 2005 de la UCLV.
- Monografía: Generación de un medio con partículas de posición arbitraria. “Método de elementos discretos o de partículas”, Editorial CIMNE. CIMNE-UCLV, Julio 2005.
- Simulación de problemas de desgaste, en la interacción herramienta de corte- terreno, empleando el método de los elementos discretos. Revista Ingeniería Civil, 141/2006, España.

Capítulo 1: Estado del arte.

1.0 - Estado del arte de la modelación de medios discontinuos.

1.1 - Introducción.

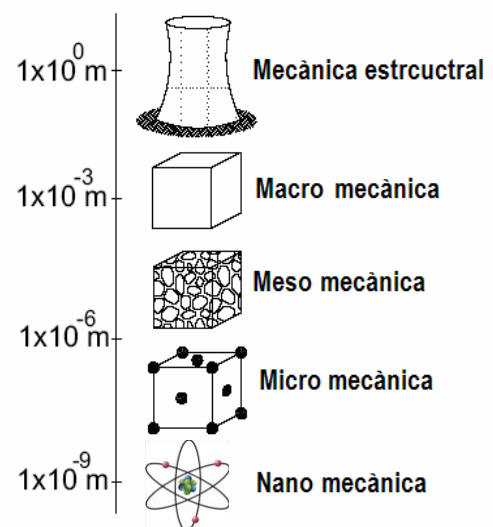
Tradicionalmente, la ciencia de los materiales estudia el comportamiento de los materiales a diferentes escalas con el objetivo de observar y cuantificar los procesos químico-físicos a niveles micro-mecánicos, moleculares y atómicos. La modelación multi-escala del material se encarga de representar estos procesos hasta el nivel microscópico. El primer paso consiste en descomponer el espectro en varios sub-rangos. Desde el punto de vista de la caracterización y del diseño de materiales ingenieriles, se distinguen las siguientes cuatro escalas:

➤ **Nivel métrico:**

A este nivel se resuelven problemas prácticos de ingeniería civil, mecánica y aeroespacial tales como el análisis y diseño de edificaciones, presas y de elementos de contención, elementos mecánicos, etc.

➤ **Nivel milimétrico**

La mayoría de las propiedades de los materiales se obtienen a partir de pruebas de laboratorio que se realizan a este nivel. Esto constituye la macro escala, un nivel en el cual los materiales se pueden considerar homogéneos después de haber homogeneizado el efecto de los elementos micro-estructurales en dichas propiedades “efectivas”.



➤ ***Nivel micrométrico***

Procesos micro-estructurales tales como micro-defectos, el tamaño de grano de polí-cristales y los productos de la hidratación de materiales cementosos se observan a esta escala. En la terminología corriente, esto constituye la meso-escala, un nivel en el cual se pueden considerar los materiales como compuestos heterogéneos. Ejemplos de esto son la matriz del metal y la mezcla de hormigón, en las cuales las partículas se unen a la matriz mediante interfases cohesivos/fricciónales.

➤ ***Nivel nanométrico***

Procesos moleculares y atómicos tienen lugar a este nivel. Varios mecanismos de difusión de humedad y de compuestos químicos agresivos se consideran activos a este nivel (transporte de iones y compuestos químicos,...). Se debe señalar que las relaciones causa-efecto llegan en algunos casos mas allá de la mecánica Newtoniana, especialmente cuando procesos sub-atómicos se consideran a nivel de la mecánica cuántica.

Generalmente, los materiales ingenieriles se consideran microscópicamente homogéneos y con frecuencia isotrópicos. En la mayoría de los casos son adecuadas estas suposiciones, ya que los procesos de degradación progresiva solo pueden ser explicados considerando características micro-estructurales del material. De ahí la necesidad de caracterizar cada constituyente y las condiciones de enlace en las interfases, además de la morfología de las meso y microestructuras específicas.

Estos tipos de estudios son muy complejos y costosos computacionalmente, a pesar del desarrollo alcanzado por los medios de cómputo y de la búsqueda de algoritmos eficientes para lograr rapidez en las simulaciones numéricas que se emplean para investigar estos procesos a niveles micro, meso, y macroscópicos en tres dimensiones y en el dominio del tiempo.

1.2 - Generalidades de la simulación numérica de medios discontinuos.

Se pueden considerar dos grandes campos de estudio en el análisis de los medios discontinuos: por una parte los métodos de cálculo para simular dichos medios y por otra la creación de modelos constitutivos para las juntas o interfases. Dada la complejidad de estos problemas es necesario recurrir a métodos aproximados para la resolución de los mismos. Los métodos de cálculo más empleados son el Método de los Elementos Finitos (MEF), el Método de los Elementos de Contorno (MEC), el Método de los Elementos Distintos (MED) o métodos que incluyen combinaciones de los anteriores.

Los métodos ideados originalmente para el análisis de medios continuos se han extendido para tener en cuenta las discontinuidades. En el MEF se consideran elementos especiales para representar las juntas o interfase. En este método otra vertiente de trabajo ha sido el desarrollo de elementos finitos especiales que se insertan dentro del campo de las discontinuidades fuertes y que emplean funciones de interpolación especiales. La otra vertiente de trabajo ha sido el desarrollo de la mecánica de la fractura. El MEC modela discontinuidades embebidas en un medio continuo. En cualquiera de estos dos métodos se asumen generalmente pequeñas deformaciones. Si se considera un comportamiento no lineal del material se requiere el uso de técnicas iterativas.

Las técnicas del MED son las formulaciones que mejor describen el comportamiento de los medios discontinuos. La génesis de estas técnicas fue desarrollada por Cundall [1] y estaba

enfocada a la resolución de problemas de geomecánica y discontinuidades en medios rocosos abordados con los conceptos de bloques y juntas. La tendencia actual de estas técnicas (Método de los Elementos Discretos o Distintos) está enfocada a posibilitar el estudio del micro-mundo y la esencia microestructural de los problemas de la mecánica computacional empleando modelos de partículas y micro-estructurales. Esta última tendencia de los MED permite realizar estudios de seguridad estructural con un alto grado de precisión y fiabilidad. Este aspecto posibilita prever fallos estructurales severos, al poderse detectar fallos a nivel microestructural (aparición de microfisuras y grietas, su evolución y concatenación). El hecho de poder estudiar el estado tenso deformacional existente entre partículas, posibilita detectar la aparición de microfisuras, las cuales son la génesis de la formación de micro zonas de falla que al final de su evolución y desarrollo culminan en los fallos estructurales o geotécnicos de una estructura u obra de ingeniería. Esta delimitación permite afirmar que las últimas tendencias de los MED posibilitan estudiar la evolución y desarrollo de las microfisuras y micro grietas conjuntamente con la metamorfosis de las mismas a fisuras y grietas macro estructurales.

La modelación y simulación numérica de medios discontinuos o continuos empleando el MED o una combinación de este con el MEF u otro método numérico, es uno de los temas que está de boga en el campo de la ingeniería. Originalmente, este enfoque de modelación ha estado dirigido a resolver problemas de ingeniería a una escala macro- estructural. En la actualidad, ha surgido una tendencia de estudiar las leyes del micro-mundo, por lo cual, la modelación y la simulación numérica se han extendido a una escala micro-estructural. Estas dos tendencias de trabajo posibilitan clasificar los métodos empleados para la modelación y simulación numérica en dos grandes grupos que toman en cuenta la escala o nivel de estudio (fig. 1).

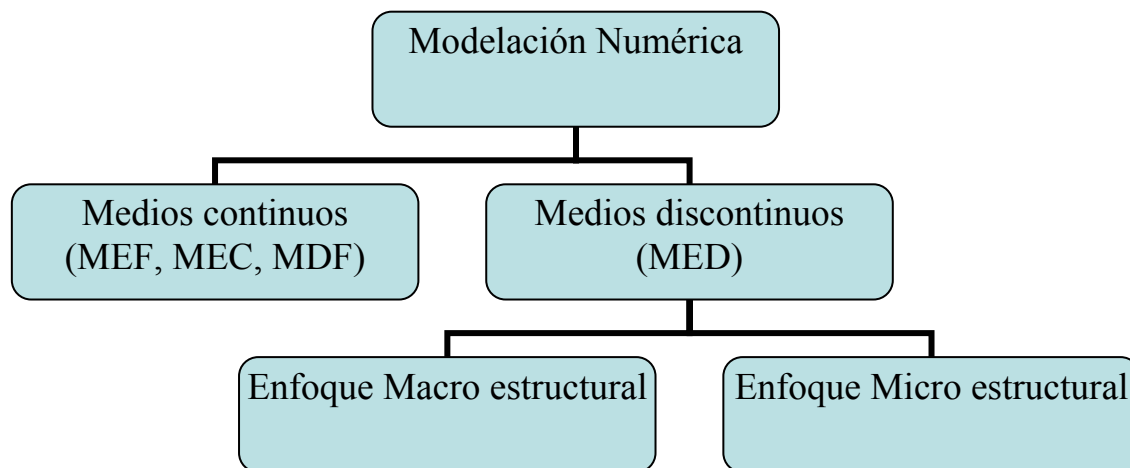


Figura 1: Tendencias de la modelación numérica.

1.3 - Generación del medio para modelación de problemas fisicomatemáticos con el método de las partículas o método de elementos discretos.

La generación del medio para resolver problemas de la física matemática con el método de partículas es un tema que se comienza a trabajar muy intensamente en el campo de la

investigación científica. Se ha desarrollado preliminarmente un grupo de formulaciones ([2] - [9]) que se encuentran en fase investigativa.

Para algunas simulaciones a través del MED, hay problemas bidimensionales en los que es conveniente utilizar una modelación a través de partículas elípticas. Chung-Yue Wangt y Vei-Chung Liangs [2] logran aproximar de forma conveniente una elipse mediante cuatro arcos de circunferencia y esbozan un algoritmo de empaquetamiento para estas partículas. En este trabajo se formula un nuevo algoritmo para generar el medio. Adicionalmente se presenta un programa de computación "**PG2D**" que logra obtener empaquetamientos de estos tipos de partículas. Estas geometrías logradas con estas formulaciones son introducidas para la simulación numérica 2D compuesta por partículas elípticas de diferentes excentricidades. Cada partícula elíptica es aproximada mediante cuatro arcos conectados. Las coordenadas del centro y el radio de cada arco y las coordenadas de los puntos de conexión pueden ser determinadas de la fórmula derivada de entrar la longitud del eje mayor $2a$, y la excentricidad. El dominio a ser llenado con partículas puede ser un polígono de cualquier forma u otra configuración geométrica lograda a través de contornos de este tipo. Dado el tamaño del dominio de empaquetamiento, información geométrica y número de partículas a ser generadas, la localización de cada partícula y las coordenadas de los puntos de contacto junto con un diagrama de las normales de contacto pueden ser generados como salida.

Los resultados de la simulación muestran que este nuevo algoritmo [2] puede proveer un modelo de empaquetamiento bastante razonable en concordancia con la entrada inicial requerida para el análisis de la mecánica de material granular. Este esquema de generación tiene el potencial de cubrir generación de empaquetamiento y análisis del comportamiento 3D de esferas (como caso particular) o materiales granulares elípticos".

En la formulación presentada en [2] hay una normal exterior única, y no hay singularidad en ningún punto en la frontera de esta elipse aproximada mediante cuatro arcos. Este tipo de aproximación puede mantener eficientemente los rasgos de elipse y provee una forma matemática simplificada al análisis asistente en realizar análisis de contactos de conjuntos de partículas de forma elíptica. Basado en esta característica, puede ser logrado una nueva formulación del análisis del comportamiento mecánico de conjuntos de partículas elípticas; o puede ser adoptado en modelos existentes desarrollados por otros investigadores como una alternativa a los esquemas de detección de contacto.

La configuración de empaquetamiento generada puede ser usada como el estado inicial para el análisis numérico del comportamiento de materiales granulares. Los autores plantean que este método puede ser fácilmente extendido a la generación de empaquetamiento de materiales tridimensionales esféricos o elipsoidales, aunque esto no es evidente.

Algunos autores como K. Nandakumar [3] empleando métodos de optimización continua logran obtener empaquetamientos de cuerpos de cierta complejidad tales como anillos de Rasching, anillos de Pall, y minianillos de cascada. Usando los avances de la geometría computacional y algoritmos de detección de colisiones se ha formulado [3] un algoritmo para analizar y predecir las propiedades geométricas de una estructura aleatoriamente empaquetada usando objetos de forma arbitraria. La dinámica del proceso de empaquetamiento no es simulada, pero un algoritmo de búsqueda encuentra posiciones de equilibrio estables para los objetos de empaquetamiento de

formas arbitrarias usando un algoritmo de detección de colisiones en un espacio 3-D. El método de optimización del gradiente conjugado modificado es el empleado para determinar la posición final del objeto o partícula, además de determinar la posición y orientación final de las mismas. Una vez que el cuerpo está empaquetado, las cantidades macroscópicas como la porosidad global, el área de superficie específica, el número de objetos de empaquetamiento por unidad de volumen y propiedades microscópicas como la variación de porosidad en cualquier dirección se pueden determinar. Para el cálculo exacto de porosidad dentro de un espacio muestral poligonal de 3-D, el algoritmo recortador de polígonos de Sutherland-Hodgman fue usado. Los resultados predichos son validados contra datos experimentales disponibles para esferas, anillos de Rasching, anillos de Pall, y minianillos de cascada.

El algoritmo de empaquetamiento de esta formulación [3] es el siguiente:

1. Seleccionar una capa de posiciones (x, y, α, β) iniciales para el elemento de empaquetamiento de manera tal que los elementos de empaquetamiento están cerca unos de otros.
2. Encontrar la posición más estable del elemento de empaquetamiento individual mediante la aplicación del método del gradiente conjugado modificado para cada elemento de empaquetamiento simple.
3. De este conjunto escoger el elemento que está más cerca de la pared del contenedor.
4. Agregar el elemento de empaquetamiento recientemente ubicado a la estructura de datos de la superficie conjunto de elementos de empaquetamiento y también borrar los elementos de empaquetamiento en la superficie que ya no representan uno de la parte superior. Este proceso es repetido hasta que un número especificado de elementos de empaquetamiento está ubicado en el contenedor. La salida de la simulación es un conjunto de posiciones finales y orientaciones (x, y, z, α, β) en el sistema global de coordenadas de objetos de empaquetamiento. Estos serán usados en subsecuentes análisis tales como la porosidad y cálculos de área de superficie.

En la investigación desarrollada por K. Nandakumar [3] además de un algoritmo se ha desarrollado un programa de computación para simular la construcción de un empaquetamiento aleatorio en un contenedor cilíndrico. La mejora más significativa sobre previos estudios es la habilidad de manipular objetos de empaquetamiento de formas arbitrariamente complejas.

La investigación del empaquetamiento de partículas busca generar conjuntos de cuerpos de formas lo más generales posibles de una manera eficiente. Caroline Hogue [4] presenta para esto variantes muy convenientes de representación de una gran clase de formas.

Los modelos que usan elementos discretos requieren información detallada de contactos individuales partícula-partícula (Cundall y Strack, 1979; Drake y Shreve, 1986; Hogue y Newland, 1994; Walton, 1983). Aunque la mayoría de los modelos son solo bidimensionales, estos también tienden a ser limitados a modelar partículas circulares o elípticas. Esto es principalmente debido a la complejidad adicional, y por lo tanto requerimientos computacionales adicionales, lo cual es más notable cuando se corren simulaciones en sistemas de media a larga escala, i.e., de cientos a miles de partículas por simulación.

Como resultado, muchos modelos computacionales no pueden abordar directamente los problemas que surgen cuando se manejan conjuntos de partículas arbitrarias (especialmente mezclas que contienen cuerpos de tipo alargado). Más aún, cuando se extienden los modelos a tres dimensiones, el problema de detección de colisiones de formas arbitrarias es grandemente acentuado.

En estas investigaciones [4] compara varios descriptores de forma en dos y tres dimensiones y sus respectivas técnicas para detección de contactos. El objetivo es encontrar descriptores de forma flexibles que puedan representar una variedad de formas dentro de una misma simulación, a la vez que permitan realizar una eficiente y precisa detección de colisiones entre cuerpos adyacentes.

Un aporte importante de este trabajo [4] es la representación de cuerpos a través de supercuádricas, las cuales son una generalización de la elipse. Su gran ventaja consiste en que mediante una misma ecuación se pueden representar muchos tipos de partículas solamente variando según se desee los parámetros de la funciones. Además, la ecuación cartesiana de las supercuádricas hace muy cómodo el chequeo de contactos si se combina con la representación discreta expuesta en el mismo artículo [4].

En la fig. 2 se muestra la gran cantidad de formas que se puede obtener con una misma ecuación variando sus parámetros:

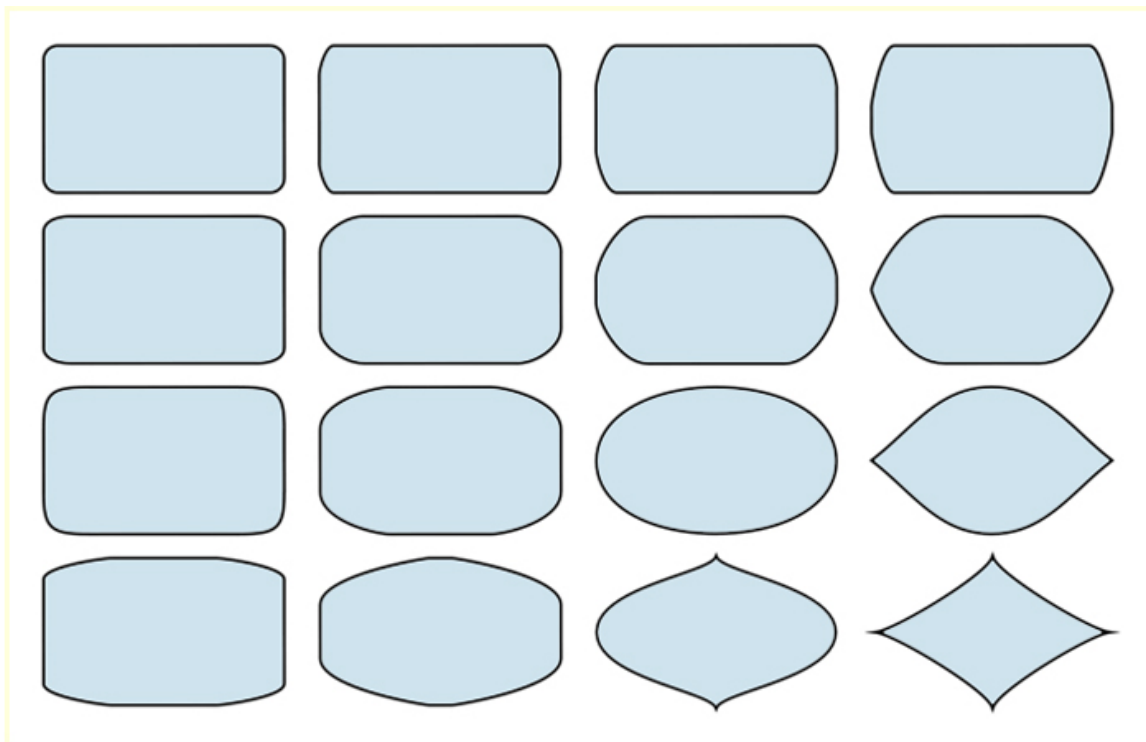


Figura 2. Supercuádricas bidimensionales.

En esta investigación [4] se revisan las representaciones actuales de formas y sus algoritmos asociados de detección de contactos en dos y tres dimensiones. También propone una extensión

de la eficiente representación polar discreta a un descriptor polar esférico 3D, como una alternativa más flexible a poliedros, y aún manteniendo la simplicidad de implementación en una simulación.

En S. Mohammadi [5] aparece la forma de representar y de detectar contactos entre partículas circulares, clusters de esferas, partículas elípticas y supercuádricas.

El chequeo de contacto entre partículas circulares y clusters de esferas es inmediato. El contacto entre dos elipses se chequea resolviendo un problema de optimización. Por último, para el chequeo del contacto entre supercuádricas se sugiere usar la propia función para representar estos cuerpos, la cual permite saber rápidamente cuándo un punto está dentro, cuándo está sobre y cuándo está fuera de la superficie de la supercuádrica.

Aunque los polígonos y poliedros son utilizados exitosamente en simulaciones de mecánica de rocas, la flexibilidad y eficiencia de representaciones de forma discretas (combinada o no con una representación continua) provee una alternativa atractiva para una clase más amplia de simulaciones.

Las técnicas de avance frontal tienen una gran aplicación para resolver problemas de ingeniería, entre los cuales se encuentran el mallado de un dominio y el empaquetamiento de partículas. Y. T. Feng, K. Han y D. R. J. Owen [6] abordan este último problema desde la perspectiva del avance frontal.

En las investigación [6] se desarrolla un algoritmo basado en técnicas de avance frontal. En este trabajo se pretende generar constructivamente un empaquetamiento aleatorio inicial para discos con radios diferentes dentro de un dominio 2D, el cual es usualmente requerido en métodos de elementos discretos (MED). Dependiendo del tipo de frontera del dominio se han formulado dos variantes del algoritmo (formas abierta y cerrada). La forma abierta tiene una complejidad lineal inherente. La forma cerrada puede alcanzar la misma complejidad bajo una condición relativamente débil. El empaquetamiento generado no es un arreglo óptimo globalmente pero alcanza una densidad local máxima desde el punto de vista algorítmico. El desempeño del algoritmo está ilustrado en varios ejemplos. El mayor beneficio de este desarrollo es la significativa reducción de tiempo de CPU requerido para la preparación de una configuración inicial de objetos o partículas en simulaciones del MED. Se demuestra que toma solo 3.77s para el algoritmo propuesto generar un millón de discos en una PC con un procesador de 1GHz.”

En esta publicación científica [6] se hace un resumen de algunos métodos de preparación de un empaquetamiento de discos para ser usado en una simulación del MED y luego propone otro método para lograr lo mismo, el cual como ya se expreso se basa en el avance frontal.

En esta investigación [6] se abordan algunos de los métodos existentes que explicados de forma breve son los siguientes:

- 1) Secuencialmente generar partículas determinando su centro de manera aleatoria, rechazando las que solapan a alguna ya colocada. Esto tiene la desventaja de que a medida que aumenta el número de partículas, aumenta la cantidad de rechazos, lo cual

consume una gran cantidad de tiempo y afecta la distribución seguida por los parámetros que definen la forma de las partículas.

- 2) Considerar inicialmente el dominio lleno de discos con radio igual al máximo, y en una segunda etapa disminuir los radios de los discos y acercar las paredes del marco que los contiene para de esta forma acercar los discos entre sí.
- 3) Dejar caer hacia el interior de un silo varias capas de discos, donde cada capa contiene cierta cantidad de discos iguales. Esto se hace usando las mismas técnicas del MED. Esta variante de empaquetamiento es muy costosa y ineficiente desde el punto de vista computacional.

Los dos últimos métodos consumen una gran cantidad de tiempo, por lo que Y. T. Feng, K. Han y D. R. J. Owen [6] propone un método puramente geométrico basado en técnicas de avance frontal. La idea general de este último consiste en ir construyendo dinámicamente un frente de discos, a los cuales se les va agregando nuevos discos a medida que se van generando. Dicho frente se actualiza de forma dinámica. La existencia de este permite en todo momento tener que considerar solamente una pequeña parte del total de partículas. Como resultado de esta investigación [6] se puede expresar que la variante que contiene la forma cerrada puede alcanzar la misma complejidad lineal bajo la condición de que un número máximo de discos en el frente sean chequeados para el posible solapamiento. Aunque el empaquetamiento generado no es un arreglo globalmente óptimo, este ha alcanzado una densidad local máxima desde el punto de vista algorítmico y debe ser suficiente para representar una situación real, como se ilustra en los ejemplos presentados [6]. La versión de la forma abierta es de un interés más práctico, ya que el espacio entre los discos y la frontera del dominio es considerablemente reducido en comparación con la forma cerrada.

La compresión y compactación adicionales pueden ser aplicadas para ajustar el empaquetamiento si una mayor densidad es requerida en la práctica. Realmente en muchos casos la fase inicial de cómputo del MED puede actuar como un medio de posterior ajuste.

El mayor beneficio de este desarrollo [6] es la significativa reducción de tiempo de CPU requerido para la preparación de una configuración inicial de objetos discretos en simulaciones del MED. Está demostrado que el presente algoritmo solo toma poco tiempo de computo para generar un millón de discos en una PC con un procesador de 1 GHz, mientras usando un silo para mezclar los discos puede tardar varios días para lograr el mismo objetivo.

Una formulación con el mismo enfoque de [6] aparece en Hernández Ortega [7]. En este caso esta última formulación [7] es análoga a la forma abierta de [6]. La formulación en dos dimensiones explica cómo llenar con discos un polígono convexo con agujeros convexos. El radio de estos discos sigue una distribución probabilística cualquiera, aspecto que le imprime una singular importancia en el proceso de generación o empaquetamiento de partículas.

El algoritmo [7] consta de tres etapas. En la primera de ellas se resuelve el problema de llenado para un dominio rectangular que contiene al dominio original. Esta es la etapa de generación propiamente dicha. La segunda etapa parte de los datos de la primera (centro y radio de cada esfera) y consiste en extraer los esferas o discos que no están dentro del dominio original (fase de eliminación) y posteriormente genera nuevas esferas para “rellenar” los huecos que han quedado entre el contorno y las esferas o discos de dentro del dominio generado. La tercera y última etapa

parte de los datos obtenidos en la segunda y sigue el mismo proceso, esto es, identifica aquellas que están dentro de los agujeros del dominio original, las extrae y posteriormente genera nuevas esferas para completar los espacios.

La colocación secuencial de las esferas en tres dimensiones es una tarea mucha más compleja que en dos dimensiones. No se describe en [7] una formulación matemática detallada, como se hizo en el caso bidimensional. Este tema presenta gran número de singularidades, sobre todo para distribuciones de radio muy dispersas. Abordar toda la casuística es una dura tarea de programación, que si bien no presenta ninguna dificultad conceptual, sí exige muchas horas de trabajo. En [7] lo que se ha pretendido es demostrar que, en los aspectos más básicos, la algoritmia es la misma.

Para lograr un empaquetamiento inicial de objetos discretos en el MED, existen otras variantes que no son necesariamente la constructiva y que también son eficientes. En el artículo de Löhner y Oñate [8] se hace una exposición al respecto. En este trabajo [8] se desarrolla una técnica de avance frontal para el llenado del espacio con objetos arbitrarios. La entrada necesaria consiste en la especificación de la distancia media puntual en el espacio y una triangulación inicial de la superficie. En esta formulación cada objeto es quitado del frente activo uno a uno y, si es posible, rodeado por nuevos objetos admisibles. Esta operación es repetida hasta que no queden objetos activos. En esta investigación dos técnicas para obtener empaquetamiento máximo son discutidas: 1- colocación de objeto más próximo (durante la generación) y 2 – mover y agrandar (después de la generación). Diferentes patrones de deposición pueden ser logrados mediante la selección del orden en el que los objetos son eliminados del frente activo. Los tiempos muestran que para objetos simples como esferas el esquema es considerablemente más rápido que con generadores de volumen de malla basados en la técnica del avance frontal, haciendo posible generar grandes ($> 10^6$) y aun óptimas nubes de puntos en cuestión de minutos en una PC. Para objetos más generales, el desempeño puede decaer dependiendo en la complejidad de los chequeos de penetración. En esta publicación [8] algunos ejemplos están incluidos, aspectos que demuestran las capacidades de la técnica desarrollada.

En este trabajo [8] se presenta un esquema que permite la generación directa de nubes de objetos arbitrarios con el mismo grado de flexibilidad que los generadores de malla no estructurados. La distancia media entre objetos (o, equivalentemente, la densidad del material) es especificada por medio de cuadrículas de fondo, fuentes y densidad adjuntadas a entidades de CAD. Para no generar objetos fuera del dominio computacional, se asume una triangulación inicial de la superficie que es compatible con la distancia media entre objetos especificada por el usuario. Comenzando desde este “frente” de objetos, nuevos objetos son agregados, hasta que ningún otro pueda ser introducido. Mientras que la técnica de avance frontal para la generación de mallas de volumen elimina una cara a la vez para generar elementos, el presente esquema elimina un objeto a la vez, intentando introducir cuantos objetos sea posible en su vecindad inmediata.

En esta trabajo [8] se realizan varios ejemplos y se realiza un estudio del tiempo de cómputo empleado en cada caso. Uno de los casos de empaquetamiento, la generación de 100 000 esferas en el cubo unidad, requirió de 33s y 120s respectivamente para distintas condiciones de densidad. Otros ejemplos han sido realizados, en este caso se trata de dos estructuras perteneciente a una obra de ingeniería. La discretización de la primera estructura se efectuó mediante 41883 esferas, lo cual computó un tiempo de 127s, con una razón de llenado de volumen de $v = 0.495$. El

llenado del silo con objetos formados por cuatro esferas computó un tiempo de 7s, para un número total de 2124 objetos, es decir, 8496 esferas. Para llenar el mismo silo con 2794 objetos formados por cinco esferas cada uno se consumió un tiempo de 10s.

Aun cuando el empaquetamiento de círculos es muy útil en simulaciones del MED, este no es su único beneficio. Xiang-Yang Li, Shang-Hua Teng y Alper Üngör [9] muestran cómo puede ser empleado eficientemente en la generación de mallas para el MEF. Un paso clave en el método de elementos finitos es generar una malla de alta calidad que sea lo más pequeña posible para un dominio de entrada. Varios métodos de mallado y heurísticas han sido desarrollados e implementados. Métodos basados en el avance frontal, triangulaciones de Delaunay y quadtree-octrees están entre los más populares. El avance frontal usa estructuras de datos simples y es eficiente. Desafortunadamente, en general, este no provee ninguna garantía del tamaño y calidad de la malla que produce. Por otra parte, los métodos de Delaunay basados en el empaquetamiento de círculos generan una malla bien formada cuyo tamaño está dentro de un factor constante de la óptima. En este artículo [9] se presenta un algoritmo de mallado, el cual combina las fortalezas del avance frontal y el empaquetamiento de círculos. En el mismo [9] se prueba que este genera una malla 2D de alta calidad, y el tamaño de la malla está dentro de un factor constante de la óptima.”

1.4 – Formulaciones y Método de Elementos de Partículas.

Los métodos de partículas o método de elementos discretos son formulaciones que resuelven múltiples problemas de ingeniería. En ellos se emplean diversos tipos de partículas y las más generalizadas son las esféricas por su simplicidad geométrica y por la forma en que se pueden detectar las vecinas y el contacto entre ellas. En función del tamaño de las partículas es que se define la escala de trabajo desde la mecánica estructural hasta la micro-mecánica. Es por eso que estos métodos se clasifican según su aplicación en dos grandes grupos: *Macroestructurales* y *Microestructurales*, aspecto que lo define el tamaño de las partículas que conforman el medio.

1.4.1 – Método de partículas con enfoque macroestructural.

En el enfoque macroestructural del MED, cada cuerpo geométrico de la simulación tiene un símil físico. Este enfoque se emplea básicamente para modelar medios discontinuos, donde cada cuerpo de la simulación representa una partícula de un material granular.

En el año 1971 *Peter Cundall* [1] como ya se hizo referencia, presentaba un método novedoso para modelar los medios rocosos discontinuos. Este método tenía la particularidad de poder simular grandes desplazamientos y rotaciones de los bloques de roca, así como separaciones entre los mismos. Básicamente era un método de relajación dinámica, que utilizaba un esquema en diferencias finitas centrado en el tiempo para integrar las ecuaciones de movimiento, las cuales venían dadas por la segunda ley de Newton. El método era válido para problemas en dos dimensiones y los bloques de roca se consideraban rígidos.

Otros autores habían tratado el problema de los medios rocosos fracturados, como *Goodman* [10], *Zienkiewicz* [11] y *Burman* [12]., utilizando el método de los elementos finitos, pero éste sólo permitía considerar pequeños desplazamientos entre los bloques.

En el año 1978 *Cundall* [13] presentó un trabajo en el que se ampliaba el modelo del año 1971 [1] a bloques de deformación constante y a bloques deformables. El modelo de bloques deformables se basaba en un trabajo de *Wilkins* [14] que consistía en utilizar un código en diferencias finitas explícito en el tiempo para determinar la deformación interna de los bloques. En este trabajo da el nombre de Método de los Elementos Distintos (MED) al método matemático para modelar el comportamiento de los bloques.

La formulación desarrollada por Cundall estaba enfocada a la resolución de problemas de geomecánica y discontinuidades en medios rocosos abordados con los conceptos de bloques y juntas, aspecto que delimita que esta técnica aborde los problemas con un enfoque macroestructural. *Cundall* [1], hace uso de una aproximación que es opuesta a la de los métodos continuos, que consiste en representar el sistema como un ensamblaje de bloques discretos. El movimiento de los bloques resulta de la propagación a través del sistema de una perturbación aplicada en la frontera. Este es un proceso dinámico donde la velocidad de propagación de la perturbación (cargas externas, desplazamiento de las fronteras, etc....) es función de las propiedades físicas de dicho sistema. El comportamiento dinámico es descrito numéricamente por medio de un algoritmo de pasos de tiempo, en el cual el tamaño del paso es elegido de manera tal que las velocidades y aceleraciones se puedan considerar constantes dentro de dicho paso de tiempo.

El MED se basa en el concepto de que si el paso de tiempo es suficientemente pequeño, durante el mismo las perturbaciones no pueden propagarse entre un elemento y sus vecinos. La restricción del paso de tiempo se aplica tanto a contactos como a bloques.

Los cálculos que se llevan a cabo en el método de los elementos distintos alternan entre la aplicación de una ley de fuerza-desplazamiento en los contactos y la segunda ley de Newton en los bloques. La ley fuerza-desplazamiento se usa para establecer las fuerzas en los contactos a través de los desplazamientos (desplazamientos lineales y giros o rotaciones). La segunda ley de Newton proporciona el movimiento de los bloques debido a las fuerzas que actúan sobre el mismo. Si los bloques son deformables el movimiento es calculado en los nodos de la malla, constituida por triángulos de deformación constante. Esta formulación satisface las leyes de conservación de la cantidad de movimiento y de la energía, como consecuencia de la aplicación exacta de las leyes de Newton. El método usa un algoritmo de pasos de tiempo para establecer la integración de las ecuaciones de movimiento del sistema. Este esquema permite un tratamiento eficiente de los fenómenos que tienen lugar en las juntas, tales como el deslizamiento y la separación. También permite modelar los sistemas sometidos a grandes cambios geométricos.

En lo que se refiere a las juntas, la influencia de las mismas en el comportamiento mecánico de las masas de material ha sido ratificada por muchos experimentos y estudios numéricos [15]. Las juntas son generalmente una fuente de deformación del sistema, porque reducen la resistencia de la masa de material y, en muchos casos, el mecanismo de fallo principal viene definido por las mismas. A menudo se pueden identificar conjuntos de discontinuidades paralelas, las cuales introducen anisotropía en la masa. La continuidad de la junta también es un parámetro importante: conjuntos de juntas con un grado elevado de continuidad dividen a la masa en una estructura de bloques que puede tener un comportamiento mecánico esencialmente distinto del correspondiente al continuo.

Básicamente existen dos tipos de aproximación para la modelación numérica de sistemas fracturados. La primera aproximación se basa en modelos continuos con relaciones constitutivas que reflejan el comportamiento del material fracturado. En la segunda aproximación las discontinuidades son representadas explícitamente en el modelo numérico.

En todos los modelos numéricos el tipo y el propósito del análisis determinan el detalle que debe ser incluido en el modelo. Si la separación de las juntas es muy pequeña comparada con el tamaño de la estructura puede ser más apropiado un modelo continuo con unas relaciones constitutivas adecuadas. Las fracturas de gran tamaño requieren una representación explícita. En muchos casos no se puede reproducir el número total de bloques, entonces se debe trabajar con modelos de menor número de bloques que puedan reflejar la deformación fundamental y el mecanismo de fallo del sistema.

Referente a las leyes constitutivas para juntas, es imposible citar todos los trabajos existentes y continuamente aparecen nuevas publicaciones, pero todos ellos buscan, básicamente, reproducir el efecto de las rugosidades en las juntas y el daño que sufren como consecuencia de los esfuerzos a que están sometidas los bloques.

Cundall y Strack [15] desarrollaron un modelo numérico capaz de describir el comportamiento mecánico de un ensamblaje de discos o de esferas. El método se basa en el uso de un esquema numérico explícito en el cual la interacción entre partículas se monitorea contacto por contacto, y el movimiento de las partículas se modela partícula por partícula. Se valida el método comparando los resultados obtenidos mediante el programa BALL con los correspondientes resultados obtenidos mediante un análisis foto elástico. De esta comparación se concluye que el método de los elementos distintos así como el programa BALL son herramientas válidas para la investigación sobre el comportamiento de materiales granulares.

Cundall [16] ha desarrollado un algoritmo al que llamó UDEC (Universal Distinct Element Code), para la modelización de los macizos rocosos fracturados en dos dimensiones. El autor [16] propone un algoritmo para calcular los factores de amortiguamiento del movimiento de los bloques. El amortiguamiento va ajustándose a lo largo de la ejecución del algoritmo, de forma que dicho amortiguamiento es proporcional a la velocidad de cambio de la energía cinética del sistema. De esta manera se diferencia, en lo que a amortiguamiento se refiere, entre los bloques que están en equilibrio o los que están libres, de aquellos que están en contacto con otros.

Cundall, Drescher, y Strack [17], usando el programa BALL, implementan el método de los elementos distintos para simular experimentos bajo una variedad de condiciones de contorno. Cantidades discretas como las fuerzas de contacto, el deslizamiento en los contactos, las velocidades y rotaciones de las partículas, así como cantidades medias tales como el estrés y las tensiones pueden ser obtenidas tanto gráficamente como numéricamente.

En este trabajo se encuentra una descripción de diferentes tipos de condiciones de contorno que se implementan en el programa BALL, una lista de mecanismos que se observaron en diferentes experimentaciones llevadas a cabo mediante el uso del BALL, así como la medición e interpretación de cantidades continuas para un ensamblaje de partículas deformadas bajo tensiones grandes.

Goodman y Shi [18] desarrollaron una teoría de bloques, basada en la proyección estereográfica, en la cual los bloques se consideran homogéneos y rígidos. El objetivo fundamental de esta teoría es el de analizar geométricamente la distribución de los distintos tipos de bloques a partir de las superficies de discontinuidad, y el de establecer en qué condiciones dichos bloques pierden la estabilidad desde el punto de vista cinemático, con el fin de resolver problemas ligados con la definición de trayectorias óptimas de excavación, las orientaciones o buzamientos de los taludes, el diseño de planos de corte de los bancos de cantera, etc.

Cundall [19], después de analizar distintos tipos de amortiguamiento de las ecuaciones de movimiento de los bloques llega a la conclusión de que algunos de los problemas que conllevan el amortiguamiento proporcional a la masa, pueden resolverse usando un amortiguamiento proporcional a la fuerza desequilibrada en los nodos (centro de gravedad de los bloques rígidos). *Lemos* [20] presenta un modelo de elementos distintos para el análisis dinámico de las rocas fracturadas, y lo aplica al estudio de los cimientos de una presa sometidos a un terremoto y al deslizamiento en las fallas por efecto de las voladuras.

Shi y Goodman [21] desarrollaron un modelo en dos dimensiones para el análisis de las tensiones, deformaciones y desplazamientos de los bloques de roca. Los bloques eran de deformación constante y se analizaban por un método similar al de los elementos finitos, en el que los elementos son los propios bloques. El acoplamiento entre bloques se hace por medio de muelles, como en el caso del método de los elementos distintos.

Brady y Lorig [22] desarrollaron códigos en diferencias finitas para simular el refuerzo en rocas mediante dos modelos distintos de refuerzo, uno basado en la acción local del elemento de refuerzo en las proximidades de una junta y otro considerando el efecto del refuerzo a lo largo de su extensión. El modelo espacial es aplicado a una excavación circular en una roca y a una pared de un talud.

Cundall y otros [23] presentan una formulación tridimensional del método de los elementos distintos materializada por el uso de una computadora personal mediante el código 3DEC. Esta formulación se basa en un algoritmo de solución dinámico en el dominio del tiempo que resuelve las ecuaciones de movimiento de un sistema de bloques tridimensionales según un esquema explícito de diferencias finitas.

Este esquema se presta bien a los estudios de mecánica de los suelos para determinar el fallo de una masa discontinua de suelo bajo la influencia de determinadas fuerzas.

No obstante se hace referencia a la necesidad de fijar un escalón de tiempo menor que cierto valor crítico que impediría la estabilidad numérica del modelo.

También se hace referencia al uso del “amortiguamiento adaptativo” que ajusta la viscosidad de forma tal que la energía absorbida por el amortiguamiento sea siempre una parte constante de la variación de energía cinética en el sistema. De esta forma, cuando la energía cinética del sistema se acerca a cero, la fuerza de amortiguamiento también tiende a anularse.

Varios ejemplos se presentan en este trabajo para demostrar la aplicación del 3DEC a problemas de mecánica de las rocas: estudios de estabilidad de taludes y de techos de túneles.

Ghaboussi [24] formula un método mixto entre el MED y el MEF, para el análisis de los sistemas de bloques en dos dimensiones. Cada bloque se considera como un elemento isoparamétrico con ocho grados de libertad (dos por cada nodo), sobre el que se aplica el MEF. El acoplamiento entre bloques no se hace como en el MEF, considerando continuidad entre los elementos, sino que los elementos se ensamblan por medio de un conjunto infinito de muelles distribuidos de forma uniforme entre cada lado de contacto y en dirección perpendicular a éstos. La ventaja de este método frente al MEF convencional es que permite grandes desplazamientos y separación entre bloques.

Cundall [25] presenta los resultados de varios ensayos numéricos realizados sobre un arreglo denso de esferas en tres dimensiones. El efecto del tamaño de la muestra se determinó en una primera serie de ensayos usando rigideces de contacto lineales. En otra serie, las rigideces de contacto se consideran no lineales, acorde con la teoría hertziana. De forma general, los resultados numéricos y físicos son en adecuación excepto para el cambio de volumen en un ensayo a tracción triaxial: esta diferencia puede indicar la presencia de consolidación inicial en la muestra física.

Brady [26] utiliza el método de los elementos distintos para el diseño de excavaciones en rocas fracturadas, considerando un modelo dinámico y un modelo en juntas que tienen en cuenta el daño que se produce en las mismas como consecuencia de los desplazamientos de cortante.

Homand [27] y otros utilizaron el programa UDEC para modelar un acantilado y compararon los resultados de las tensiones con medidas hechas “in situ”, comprobando que existía una correspondencia bi-unívoca entre el modelo y la realidad.

Swoboda y otros [28] desarrollaron una teoría para definir el daño en las juntas mediante la definición de un tensor de daño, válido para un número ilimitado de juntas. Este modelo introduce este tensor de daño en las ecuaciones del modelo continuo y resuelve el problema por el método de elementos finitos. El método es aplicado para el diseño óptimo de un túnel.

Ishida [29] presentó un trabajo que consistía en comparar los resultados obtenidos con el MED, con ensayos experimentales y con resultados obtenidos a partir de métodos analíticos para tres problemas: uno para un pilar formado por bloques situado en un plano inclinado, otro para cuatro bloques cúbicos unidos formando otro cubo sobre un plano inclinado, y un tercero que consistía en un conjunto de bloques alargados situado sobre otro plano inclinado. En este último caso se estimaba la fuerza que se debía aplicar al bloque inferior para que el resto no volcase. Los resultados obtenidos con el MED concordaban muy bien con los ensayos de laboratorio y, en los casos en que pudo aplicarse, con los métodos analíticos.

Chang y Chao [30] utilizaron el MED para el análisis de la distribución de presiones en el terreno detrás de un muro de contención para diferentes formas de movimiento del muro. El muro y el terreno se modelaban mediante bloques unidos por muelles Winkler. Los resultados obtenidos fueron comparados con el método de equilibrio límite y con el método de elementos finitos.

Pan y Reed [31] propusieron un modelo híbrido entre el método de elementos finitos y el de elementos distintos para el análisis de grandes deformaciones en masas de roca en dos dimensiones. Las rocas duras, o sometidas a bajas presiones, se modelan como bloques rígidos por el MED y las más débiles, o sometidas a altas presiones, se modelan como un medio continuo

por el MEF. El acoplamiento entre los dos dominios se hace imponiendo las condiciones de compatibilidad de los desplazamientos y de equilibrio de las fuerzas. El modelo es aplicado a la excavación de un túnel, en el que la parte superior se resuelve por el MED y la inferior mediante el MEF.

Hocking [32] utiliza el método de los elementos distintos para hacer un análisis tridimensional de un bloque de hielo, con forma de tronco de cono recto, que impacta sobre un conjunto de bloques de hielo prismáticos, sobre los cuales se considera un modelo de fractura. El modelo teórico fue validado por un experimento a escala real.

Chang y Chao [33] estudiaron la carga límite de un cimiento mediante el MED. El suelo era modelado mediante bloques sometidos a compresión conectados por muelles Winkler. Las tensiones calculadas en la frontera satisfacen las condiciones de equilibrio y no exceden la resistencia del material. Los resultados se compararon con los obtenidos mediante la ecuación general de carga límite.

Arumugan, Zaman y Laguros [34] desarrollaron un algoritmo, basado en el MED, para analizar la respuesta dinámica de un pavimento de hormigón formado por losas cuadradas cuando está sometido a la acción del movimiento de un avión. Las uniones de las placas de hormigón se modelan, en el caso de las juntas longitudinales, mediante elementos viga. La interacción entre el pavimento y las ruedas del avión se simula por medio de masas unidas al pavimento mediante muelles y émbolos. Los resultados son comparados con soluciones analíticas y con datos experimentales.

Chang [35] comparó los resultados obtenidos por el método de equilibrio límite con el MED en el análisis de la estabilidad de taludes. En el MED consideró el talud formado por un conjunto de bloques unidos por muelles Winkler. Los resultados obtenidos muestran la aplicabilidad del método de los elementos discretos al análisis de estos problemas, proporcionando valores de las tensiones y del coeficiente de seguridad inferiores a los obtenidos por el método de equilibrio límite.

Billaux y Cundall [36] proponen una nueva formulación del método de diferencias finitas para la simulación de los geomateriales, conocido como método de los elementos lagrangianos. Este método es especialmente apropiado para problemas de grandes deformaciones y comportamiento no lineales, como la plasticidad. En este método los sólidos se discretizan en cuadriláteros que se consideran, a su vez, subdivididos en dos familias de dos triángulos. En cada uno de los triángulos se plantea numéricamente el equilibrio del continuo para después interpolar sobre los dos triángulos de cada familia y sobre las dos familias (método de discretización mixta). Esta formulación es un escalón intermedio entre la modelación a escala macroscópica y la modelación microestructural.

Topping y Khan [37] presentaron varios esquemas de computación paralela para los métodos de relajación dinámica, como es el MED. Mediante estos esquemas se puede obtener un considerable ahorro de tiempo de ordenador, dividiendo el dominio del problema en subdominios, sobre los que se resuelve el problema independientemente en procesos en paralelo para, posteriormente, volver al programa raíz y acoplar los subdominios.

Williams y O'Connor [38] desarrollaron un algoritmo para la detección de contactos entre polígonos de distintas formas. El algoritmo tiene una complejidad lineal con respecto al número de vértices de los polígonos, y se combina con otro algoritmo para detectar los polígonos potencialmente susceptibles de contactar con uno dado.

Munjiza, Owen y Bicanic [39] desarrollaron un modelo mixto elementos finitos- elementos discretos para simular fenómenos dinámicos en sólidos con fracturación progresiva. La fracturación del cuerpo induce la formación de los elementos discretos, los cuales pueden estar compuestos de uno o más elementos finitos deformables. El modelo de fracturación está basado en la fractura de Rankine. También se establece un modelo que tiene en cuenta la interacción del sólido con los gases procedentes de las voladuras de rocas. El método es aplicado a algunos problemas de interés industrial, como voladuras en rocas, demolición de una chimenea o el impacto de un vehículo contra el pilar de un puente.

Krishnasamy y Jakiela [40] propusieron un método para detectar contactos entre bloques poligonales en dos dimensiones que evita el problema de la ambigüedad en la determinación de los contactos vértice-vértice, y el de las discontinuidades en las fuerzas de contacto al pasar de un tipo de contacto a otro. El método está basado en la determinación del instante en el que se produce el contacto y el instante en que desaparece.

Hassen, Spinnler y Fine [41] propusieron un método de análisis de la capacidad de las rocas para ser perforadas. Este método consta de tres etapas principales. La primera utiliza el MEF sobre el medio continuo para establecer una relación de perforabilidad entre tensiones iniciales, dimensiones de la roca y resistencia mecánica de la misma. La segunda etapa hace un estudio de la densidad de fisuración y de la distribución de los bloques. La tercera lleva a cabo un estudio del movimiento de los bloques mediante el MED.

Otsu, Mori y Osakada [42] obtuvieron las condiciones optimales de trabajo para la conformación en estado pastoso de imanes a base de lantánidos usando el método de los elementos distintos. El desplazamiento de los granos se calcula mediante el MED y la deformación plástica de la cápsula que contiene el imán se calcula mediante el método visco plástico de los elementos finitos. Se modeló la conformación de imanes rectangulares contenidos en cápsulas de acero bajo presión plana con moldes cóncavos o convexos.

Selvadurai y Sepehr [43] aplican el MED para simular la interacción de una capa de hielo en movimiento con una estructura flexible inmóvil. El MED se usa para modelar el fallo en una capa de hielo inicialmente intacta. El fallo del hielo corresponde a situaciones donde este último puede sufrir combinaciones de fragmentación frágil y fenómeno de flujo visco plástico. El modelo tiene en cuenta la variación de resistencia de los bloques de hielo después de la fragmentación, en función de su tamaño.

Han, Peric, Crook y Owen [44] en una simulación del proceso de “shot peening” donde se combinan el MED y el MEF, estudian diferentes leyes de interacción de contacto para casos en 2D con especial atención acordada a la apropiada selección de los valores de los parámetros involucrados. Además se realizan los estudios de fuerzas de contacto, contacto parcial, disipación de energía y representación reológica.

Ejemplos numéricos para un sistema con impacto de un tiro único obtenidos mediante una simulación combinando el uso del MED y del MEF usando diferentes leyes de interacción se proveen para verificar los acercamientos propuestos.

P. Cleary [45] simuló, mediante el uso del MED, la molienda de material con partículas de diferentes tamaños y densidades interactuando con objetos de formas complejas moviéndose según diferentes patrones. El comportamiento de la carga, el par motor y los requerimientos de energía se estudiaron para un rango de velocidades de rotación del 50 al 130 por ciento de la velocidad crítica para un molino de bola. Se examinó la sensibilidad de los resultados frente a las propiedades del material y a la distribución en tamaño de las partículas. Una comparación de los flujos obtenidos con la simulación de un molino centrifugador con una serie de fotografías experimentales tomadas a altas velocidades revela una buena adecuación del modelo.

Bierowski, Shiro Maeno, Hitosho Gotoh, y Eiji Harada [46] combinaron las técnicas del MED y del MEF para modelar numéricamente el colapso de una estructura compuesta de un muro vertical fijo y de la masa de arena que contiene bajo el impacto de olas en un dominio vertical. El objeto del trabajo era el análisis de los procesos que ocurren en el material granular en la vecindad del muro bajo la acción del agua. El estudio se basa en los métodos de mecánica computacional respecto al transporte de sedimentos, razón para la cual se consideró la interacción entre los sedimentos y las oscilaciones del nivel de agua libre así como las colisiones interpartículas. Ensayos de laboratorio a pequeña escala se realizaron para corroborar los resultados del modelo numérico. En el caso de esta investigación, se emplea eficientemente el enfoque de combinar técnicas de instrumentación y experimentación para la calibración de los modelos numéricos. Este trabajo se basa en estudios de varios investigadores como Yamamoto (1981), Nago (1982), Zen (1984), y es la continuación de investigaciones precedentes realizadas por Nago y Maeno (1987 y 1993), Nago et al. (1995), Maeno et al. (1999 y 2002).

Hazzard, Young y Oates [47] emplearon un modelo basado en el MED para simular la inyección de fluido en un estrato de granito. La roca se modela mediante un arreglo de partículas unidas entre sí por puntos de contacto. El flujo de fluido se simula también a micro escala. Es un modelo dinámico que permite entonces obtener información sísmica durante la deformación y la fractura. En esta investigación, se emplea el software comercial PFC^{2D} desarrollado por el grupo de consultoría ITASCA.

1.4.2 – Método de partículas con enfoque microestructural.

El enfoque microestructural del MED se emplea básicamente para modelar medios continuos. En este enfoque cada cuerpo geométrico de la simulación por lo general no tiene un símil físico. De una correcta selección de los parámetros de los elementos discretos dependerá la calidad de la simulación, con el inconveniente de que estos parámetros son de elementos discretos que no existen físicamente.

En el MED los cuerpos preferidos para las simulaciones, siempre que el problema lo permita, son las esferas debido a su simplicidad. Sin embargo, a partir de un cuerpo tan elemental como la esfera es posible obtener una gran variedad de formas construyendo conjuntos de esferas que se consideran como un cuerpo rígido. J. F. Favier [48] presenta una formulación basada en esta idea. En esta investigación [48] se presenta un nuevo método de representar partículas no esféricas, de superficie suave, axial simétricas, en simulaciones de elementos discretos usando modelos de

partículas compuestas por esferas que se solapan. El tamaño de las partículas o clusters de esferas es arbitrario y sus centros están fijos en posición relativa a cada una a lo largo del eje mayor de simetría de la partícula. La detección de contactos, el cálculo de fuerza-deformación y el movimiento de la partícula es logrado usando técnicas estándar de elementos discretos modificadas para integrar el comportamiento de cada elemento esférico con el de la partícula multi-elemento al cual este pertenece. El método permite que sea modelado el comportamiento dinámico de partículas de gran excentricidad y curvatura irregular (en dos dimensiones). El uso de esferas para representar una partícula aprovecha la velocidad computacional y precisión de la detección de contacto para esferas, lo cual debería hacer el método comparable en eficiencia computacional a esquemas alternativos para representar partículas no esféricas.”

Es cierto que el hecho de que las partículas sean axial simétricas simplifica mucho los cálculos en el momento de la simulación mediante elementos discretos, ya que de esta forma el tensor de inercia es una matriz diagonal; aunque también se cumple que esta condición no es en modo alguno necesaria para obtener el empaquetamiento inicial. Esta formulación tiene el inconveniente de que las partículas en si tiene que ser axial simétricas cuando en la realidad existen partículas que no siguen esta regularidad.

Una formulación más general sobre este tipo de partículas en el MED puede encontrarse en Itasca Consulting Group [49], donde no hay necesariamente simetría de las partículas respecto a un eje. En esta investigación [49] se define la masa, el centroide y el tensor de inercia relativo a cada partícula. Se puede notar que en esta última fórmula el tensor de inercia, en el caso general, no es simétrico, a diferencia de lo aparecido en [48].

En este trabajo [49] se aborda el tema de las ecuaciones de movimientos de un sistema de partículas con estas peculiaridades. En estas las principales herramientas matemáticas empleadas son el método de diferencias finitas y un algoritmo iterativo formulado para resolver un sistema en específico de ecuaciones no lineales. Finalmente, aparecen algunos ejemplos de simulaciones.

Los medios granulares están presentes dondequiera en la naturaleza y en varios campos de actividad humana: arena, cereales y diques son ejemplos comunes. Estos medios tienen propiedades que no se encuentran en alguno de los tres estados usuales de la materia, por ejemplo el efecto de arqueamiento, segregación y convección. La comprensión de estos fenómenos es todavía incompleta pero es enormemente aumentada mediante la simulación numérica que se ha convertido en una apreciable vía de investigación gracias al creciente poder de las computadoras modernas. Por estas razones se desarrollan cada vez más investigaciones en esta temática.

Investigadores como Müller [50] realiza una formulación bastante abarcadora del MED. En este caso el objetivo de [50] es estudiar varias técnicas computacionales para simulaciones rápidas y realísticas cuando los granos son modelados como individuos. Los autómatas celulares son la vía más simple para simular un medio granular. Discretizar el espacio y el tiempo hace el cómputo muy rápido. Sin embargo, las leyes físicas son aproximadas mediante reglas evolucionarias simplificadas que necesitan ser ajustadas para cada experimento. La interpretación de los resultados puede ser en el mejor de los casos cualitativa.

Un enfoque muy interesante, pero que requiere un gran número de operaciones computacionales es el llamado Método de Elementos Distintos o Método de Partículas. En este caso esta

formulación trata de simular los problemas a escala de la micro-estructura de los materiales. Cada grano es modelado separadamente y los contactos necesitan ser detectados para aplicar leyes físicas apropiadas. Determinar los contactos es la parte de cálculo que más tiempo consume. Para acelerar la búsqueda, es necesario usar funciones de vecindad que se aplican solamente a una subdivisión del medio granular.

Para una eficiente identificación de los puntos de contacto, en [50] se propone una triangulación adaptativa de vecindad. Algunos métodos requieren el conocimiento exacto de los tiempos de colisión; las triangulaciones permiten calcularlos. Varios tipos de granos (discos, polígonos, etc.) han sido usados y varias escuelas (partículas rígidas y suaves) han sido explotadas y comparadas. Desafortunadamente las triangulaciones no pudieron ser aplicadas al caso de tres dimensiones, por razones intrínsecas.

Los modelos de partículas suaves fueron encontrados particularmente buenos para el paralelismo y los incrementos de velocidad obtenidos al combinar esta técnica con triangulaciones son impresionantes. Las pruebas fueron realizadas en una supercomputadora Cray T3D.

Los métodos desarrollados en [50] han sido probados en simulaciones de varios experimentos: visualización de segregación y fenómenos de convección que ocurren cuando un medio granular es sometido a vibraciones, análisis de sensibilidad de varios parámetros en flujos en un silo, etc. Las ventajas de la simulación para estudiar medios granulares y la posibilidad de visualizar trayectorias, fuerzas, velocidades, etc., y el hecho de que los parámetros del experimento pueden ser fácilmente modificados le imprimen a esta formulación un alto grado de flexibilidad y versatilidad.

Alonso y Gilli [51] introducen, en un sistema de partículas distintas, fases líquidas y gaseosas para poder analizar el comportamiento de un suelo limo-arenoso con un contenido bajo de agua. Mediante el modelo, se predicen los cambios en cuanto al contenido de agua y su distribución así como el tamaño de los poros, las fuerzas de contacto y las deformaciones dadas para un estado dado de saturación, de arreglo geométrico y de fuerzas externas aplicadas. Las partículas sólidas se modelaron como esferas cuasi-rígidas. También fueron modelados los poros y el menisco de agua así como la transferencia de agua entre estas entidades. Algunos aspectos básicos de la modelación mecánica se tomaron del trabajo de Cundall y Strack [15].

Trent y Margolin [52] utilizaron el método de los elementos distintos para realizar una serie de experimentos numéricos similares a las pruebas que suelen realizarse sobre materiales geológicos reales.

Disponiendo de las propiedades exactas del material así como de los vínculos entre partículas, se pudo estudiar detalladamente la influencia de la estructura microscópica sobre la respuesta macroscópica. Se obtuvo que el coeficiente de Poisson macroscópico depende en mayor parte de la topología del vínculo. Una reducción del módulo de elasticidad del vínculo tiene como resultado, de forma general, una reducción del módulo macroscópico por el mismo factor.

Kuhn [53] describe un mecanismo de frontera flexible para modelar el ensamblaje de partículas esféricas por medio del MED. La frontera está formada por elementos triangulares conectados

por sus vértices a los centros de las esferas vecinas. Las tensiones externas aplicadas en la frontera crean fuerzas de tracción en los triángulos. Los ensayos de compresión plana en ensamblajes de esferas con fronteras flexibles dan lugar a tensiones inferiores a las que se obtienen cuando se consideran fronteras rígidas.

Cundall [54] ha desarrollado dos nuevos paquetes comerciales basados en el MED que llevan los nombres de *PFC2D* y *PFC3D* (Partible Flow Code in Two and Three Dimensions). En estos programas se consideran los sólidos en dos dimensiones y tres dimensiones formados por partículas circulares y esféricas. Estas partículas pueden representar granos individuales en un material, como puede ser el caso de la arena, o también puede considerarse las partículas fuertemente unidas para representar materiales más cohesivos como rocas u hormigón. Las partículas son rígidas pero se deforman localmente en los contactos. Las aplicaciones del código incluyen el análisis de materiales compuestos de granos, flujo de partículas en silos, estudio de la microfracturación en sólidos, colapso de taludes, cimentaciones y túneles. El modelo permite simular el movimiento y la interacción en materiales discontinuos, incluyendo deslizamiento y separación.

Tang-Tat. y Petrakis [55] modelan un suelo granular a partir de un arreglo tridimensional aleatorio de esferas de cuarzo mediante el uso del MED. La simulación tridimensional realizada con una presión baja incluye la determinación del módulo de confinamiento a nivel microscópico para arreglos de esferas cargadas isotrópicamente y anisotrópicamente. Los resultados están en adecuación con los ensayos triaxiales llevados a cabo en la Universidad de Texas. Se halló que un cambio de distribución de las fuerzas de contacto juega un papel muy importante en aquel fenómeno.

Otsu, Mori, y Osakada [56] presentan el uso del método de los elementos distintos para simular el comportamiento microscópico en el proceso de fundición en estado pastoso. En esta simulación, se tiene en cuenta la presión de la fase líquida dividiendo el sistema en varios elementos tetraédricos generados por la conexión de los centroides de los elementos distintos esféricos usando la triangulación de Delaunay. Cuando una pieza en estado pastoso se deforma plásticamente, la fase líquida fluye entre las partículas. Esto trae como consecuencia que la fracción volumétrica de fase sólida no sea homogénea. Como el flujo de la fase líquida es influido por la distribución de la presión en la pieza, se introdujo el efecto de esta en el método de los elementos distintos.

Bagi, Bojtar, y Galos [57] estudiaron la resistencia al fallo a compresión uniaxial de suelos granulares. Los resultados de los ensayos de laboratorio, generalmente, presentan una gran dispersión que suele prestarse a los pequeños errores aleatorios que se producen a la hora de realizar los experimentos. En este trabajo se supervisó este problema con un enfoque micro estructural para demostrar que la aleatoriedad geométrica de la micro estructura tiene un papel decisivo en cuanto a las diferencias entre los resultados de pruebas individuales.

Kuo-Neng, Chang y Meegoda [58] desarrollaron un programa de cómputo denominado ASBAL. Este programa tiene implementado un modelo basado en el MED y se desarrolló a partir del programa TRUBAL para simular una mezcla asfáltica en caliente. En este caso, el modelo constitutivo emplea varios elementos visco elásticos (elementos de Maxwell, Kelvin-Voigt, y Burger) obteniendo mejores resultados con el elemento lineal visco elástico de Burger. Las

simulaciones realizadas con ASBAL se aproximan satisfactoriamente con los resultados experimentales.

Gethin, Ransing, Lewis, Dutko y Crook [60] consideraron una combinación del método de los elementos finitos con el método de los elementos distintos para simular la compactación de un sistema de partículas. Cada partícula se modela mediante elementos finitos y las interacciones entre partículas se modelaron mediante la técnica de los elementos distintos. Este método es equivalente al modelo de Gurson con análisis en medio continuo ya que los resultados obtenidos a partir de los dos análisis se corroboran. El trabajo demuestra la potencial aplicación de este modelo de elementos distintos deformables al caso de la compactación de materiales granulares en generales y de polvo en particular.

Rojek, y otros autores [60] presentan un modelo numérico para rocas, suelo y medios granulares usando elementos distintos esféricos. La integración explícita en el tiempo presenta una alta eficiencia computacional. El modelo constitutivo empleado en la interfase de contacto entre las esferas toma en cuenta las fuerzas de cohesión que permiten modelar la fractura y la descohesión de los materiales. Él permite entonces describir el comportamiento de materiales granulares así como de los materiales sólidos caracterizados por un fallo frágil. Ejemplos verifican la veracidad del algoritmo desarrollado por estos autores [60] pertenecientes al Centro Internacional de Métodos Numéricos en la Ingeniería(CIMNE). En publicaciones más recientes [61], se emplea el MED para modelar el proceso de fundición. La simulación numérica predice los defectos del molde debidos a una compactación insuficiente de la arena alrededor de la pieza. El uso combinado del MED para modelar las partículas de arena y del MEF para modelar la pieza de poliespuma permite detectar una posible distorsión de la pieza durante el llenado del molde y la compactación.

1.5 – Conclusiones parciales.

Después de haber realizado una exhaustiva investigación del estado de arte de la temática de generación, simulación y modelación de problemas de mecánica computacional relacionadas con el método de partículas se ha llegado a un grupo de conclusiones que justifican el trabajo científico a realizar. En esta dirección a modo conclusivo se pueden expresar varios aspectos de las dos temáticas abordadas: generación o empaquetamiento de partículas y métodos de elementos distintos o de partículas.

1 - Generación o empaquetamiento de partículas:

- Se manifiesta la existencia de métodos de generación y empaquetamiento específicos para algunos tipos de partícula. No existe un método de generación o empaquetamiento de partículas de diversas formas con enfoque sistémico e integrador. En este caso se evidencia la inexistencia de un enfoque metodológico que posibilite por una misma formulación lograr empaquetar partículas de formas y posición arbitraria.
- En la formulación presentada en [2] hay una normal exterior única, y no hay singularidad en ningún punto en la frontera de esta elipse aproximada mediante cuatro arcos. Este tipo de aproximación puede mantener eficientemente los rasgos de elipse y provee una forma matemática simplificada al análisis asistente en realizar análisis de contactos de conjuntos

de partículas de forma elíptica. Basado en esta característica, puede ser lograda una nueva formulación del análisis del comportamiento mecánico de conjuntos de partículas elípticas; o puede ser adoptado en modelos existentes desarrollados por otros investigadores como una alternativa a los esquemas de detección de contacto.

La configuración de empaquetamiento generada puede ser usada como el estado inicial para el análisis numérico del comportamiento de materiales granulares. Se cree que este método puede ser fácilmente extendido a la generación de empaquetamiento de materiales tridimensionales esféricos o elipsoidales, aunque esto no es evidente.

- En la investigación desarrollada por K. Nandakumar [3] además de un algoritmo se ha desarrollado un programa de computación para simular la construcción de empaquetamiento aleatorio en un contenedor cilíndrico. La mejora más significativa sobre previos estudios es la habilidad de manipular objetos de empaquetamiento de formas arbitrariamente complejas.
- Un aporte importante de este trabajo [4] es la representación de cuerpos a través de supercuádricas, las cuales son una generalización de la elipse. Su gran ventaja consiste en que mediante una misma ecuación se pueden representar muchos tipos de partículas solamente variando según se desee los parámetros de la funciones. Además, la ecuación cartesiana de las supercuádricas hace muy cómodo el chequeo de contactos si se combina con la representación discreta expuesta en el mismo artículo [4].

También propone una extensión de la eficiente representación polar discreta a un descriptor polar esférico 3D, como una alternativa más flexible a poliedros, y aun manteniendo la simplicidad de implementación en una simulación.

- En la publicación científica [6] se hace un resumen de algunos métodos de preparación de un empaquetamiento de discos para ser usado en una simulación del MED y luego propone otro método para lograr lo mismo, el cual como ya se expreso se basa en el avance frontal. En [6] no se explica cómo puede extenderse el algoritmo ahí propuesto a tres dimensiones.

El mayor beneficio de este desarrollo [6] es la significativa reducción de tiempo de CPU requerido para la preparación de una configuración inicial de objetos discretos en simulaciones del MED. Está demostrado que el presente algoritmo solo toma poco tiempo de cómputo para generar un millón de discos en una PC con un procesador de 1 GHz, mientras usando un silo para mezclar los discos puede tardar varios días para lograr el mismo objetivo.

- En la formulación [7] se explica cómo llenar con discos un polígono convexo con agujeros convexos. El radio de estos discos sigue una distribución probabilística cualquiera, aspecto que le imprime una singular importancia en el proceso de generación o empaquetamiento de partículas. Sin embargo, en esta investigación [7] no se da un algoritmo explícito para lograr lo mismo en tres dimensiones.

- En la investigación [8] se presenta un esquema que permite la generación directa de nubes de objetos arbitrarios con el mismo grado de flexibilidad que los generadores de malla no estructurados. La distancia media entre objetos (o, equivalentemente, la densidad del material) es especificada por medio de cuadrículas de fondo, fuentes y densidad adjuntadas a entidades de CAD. Se presentan varios ejemplos que demuestran la gran eficiencia de los algoritmos.
- En el artículo [9] se presenta un algoritmo de mallado, el cual combina las fortalezas del avance frontal y el empaquetamiento de círculos. En el mismo [9] se prueba que este genera una malla 2D de alta calidad, y el tamaño de la malla está dentro de un factor constante de la óptima.”
- No existen publicaciones sobre un algoritmo para el empaquetamiento de polígonos, o poliedros, o de elipsoides, o supercuádricas; ya que dicho empaquetamiento se obtiene a través del propio MED, lo cual es extremadamente costoso.

2. Métodos de elementos distintos o de partículas.

- No se evidencia en la bibliografía consultada una formulación del método de partículas que posibilite realizar simulaciones y cálculos con diversos tipos de partícula. Se manifiesta la existencia de diversas formulaciones para algunos tipos de partícula específica y no se ha detectado la realización de una formulación más abarcadora en esta dirección.
- Los algoritmos de búsqueda de contacto presentes en la literatura consultada presentan una serie de ventajas e inconvenientes técnicos desde el punto de vista de costo computacional, por lo que es de utilidad el desarrollo de nuevas formulaciones que mejoren las deficiencias detectadas.
- En los métodos de partículas se emplean diversos modelos constitutivos y aparecen diferentes formulaciones, cada una con modelos de contacto diferentes o con peculiaridades puntuales en cada caso.
- Existen diferentes métodos de integración para resolver la ecuación fundamental del método de partículas. El más generalizado es el de diferencia finita central, aunque se pueden emplear otros como: diferencia finita hacia delante y hacia atrás, método de la aceleración promedio, método de Houbolt o método de Wilson.

Capítulo 2: Generación del medio.

2.0 – Introducción y aspectos generales.

En la aplicación del Método de Elementos Distintos (MED) es necesario tener un empaquetamiento o generación inicial de partículas antes de comenzar la simulación. Este empaquetamiento inicial puede lograrse a través de las propias técnicas de simulación del MED, aunque de esta manera se pierde mucho tiempo. De ahí surge la necesidad de crear los llamados algoritmos de empaquetamiento o de generación del medio, los cuales no hacen más que colocar un conjunto de partículas aleatorias en el interior de un dominio de manera que no se solapen y estén lo más cercanamente posible entre sí. En este capítulo se presenta un algoritmo con dichas características, con el cual es posible generar un medio compuesto por partículas de forma arbitraria.

Durante la formulación de este método de empaquetamiento de partículas o generación del medio se empleará la siguiente notación:

Notación:

$C(c, r)$: circunferencia de centro c y radio r

$E(c, r)$: esfera de centro c y radio r

2.1 - Detección de contactos.

La detección de contactos entre dos círculos o dos esferas se puede realizar de forma inmediata, sabiendo que el contacto ocurre si y solo si la distancia entre los centros es menor que la suma de los radios.

El caso de polígonos, poliedros y supercuádricas es algo más complejo. Uno de los métodos reportados en la literatura es el siguiente: supóngase que se tiene un cuerpo A_1 y se desea saber si está en contacto con otro A_2 . Supóngase que se tiene una malla de puntos que están sobre la frontera de A_1 y otra malla de puntos que está sobre la frontera de A_2 . Entonces se puede afirmar con cierta probabilidad que A_1 y A_2 están en contacto si y solo si algún punto de la malla relativa a alguno de ellos se encuentra dentro del otro cuerpo, donde esta probabilidad aumenta a medida que aumenta la cantidad de puntos en las mallas mencionadas. Para más detalles véase Hogue [1].

2.2 - Construcción de una circunferencia tangente a otras dos.

Considérese el problema de hallar el centro (x, y) de una circunferencia de radio r_3 , tangente exteriormente a otras dos de centro $c_i = (c_{i1}, c_{i2})$, y radio r_i respectivamente, $i = \overline{1,2}$. La tangencia entre $C((x, y), r_3)$ y $C(c_i, r_i)$, $i = \overline{1,2}$, implica que $(x, y) \in C(c_i, r_i + r_3)$, lo cual equivale a que

$$(x - c_{i1})^2 + (y - c_{i2})^2 = (r_3 + r_i)^2, \quad i = \overline{1,2} \quad (2.2.1)$$

Resolviendo el sistema anterior de dos ecuaciones en las incógnitas x, y , se obtiene la solución deseada. Nótese que en el caso general existen dos soluciones.

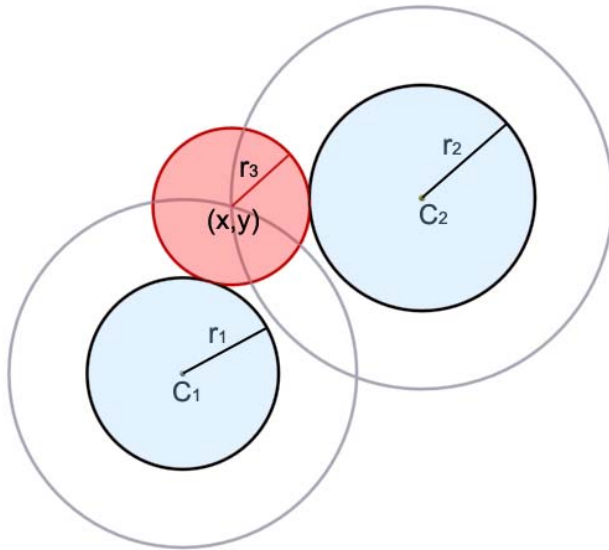


Figura 1: Construcción de una circunferencia tangente a otras dos.

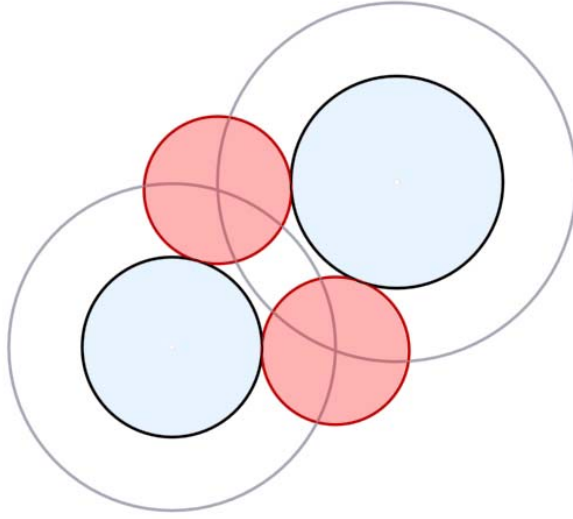


Figura 2: Representación esquemática de posibles soluciones. Se evidencia que el sistema (2.2.1) tiene dos soluciones en el caso general.

2.3 - Construcción de una esfera tangente a otras tres.

Considérese el problema de hallar el centro (x, y, z) de una esfera de radio r_4 , tangente exteriormente a otras tres de centro $c_i = (c_{i1}, c_{i2}, c_{i3})$, y radio r_i respectivamente, $i = \overline{1,3}$.

La tangencia entre $E((x, y, z), r_4)$ y $E(c_i, r_i)$, $i = \overline{1,3}$, implica que $(x, y, z) \in E(c_i, r_i + r_4)$: lo cual equivale a que

$$(x - c_{i1})^2 + (y - c_{i2})^2 + (z - c_{i3})^2 = (r_4 + r_i)^2, \quad i = \overline{1,3} \quad (2.3.1)$$

Resolviendo el sistema anterior de tres ecuaciones en las incógnitas x, y, z , se obtiene la solución deseada. Nótese que en el caso general existen dos soluciones.

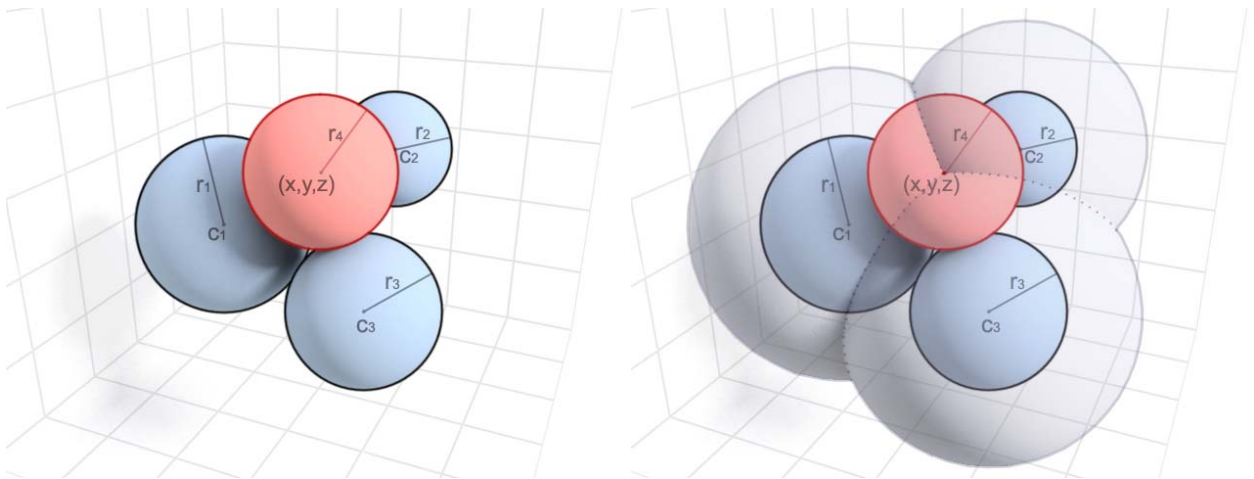


Figura 3: Construcción de una esfera tangente a otras tres.

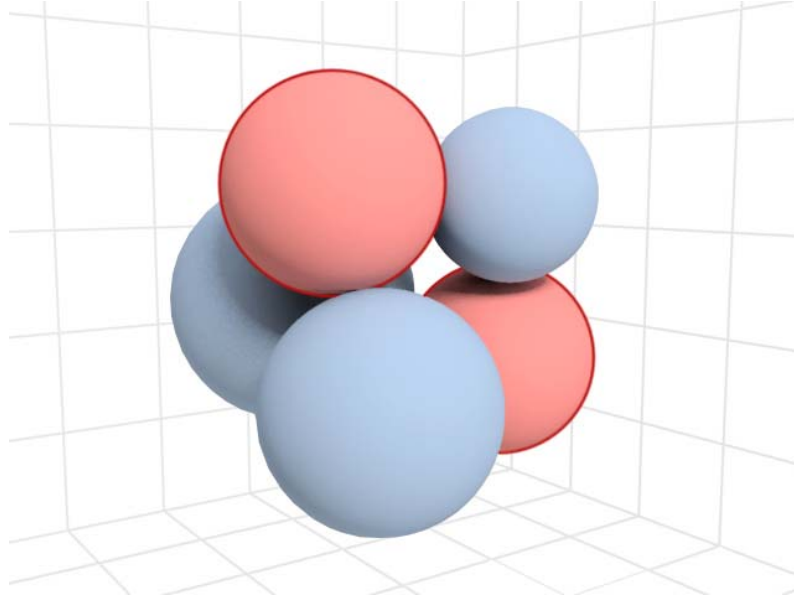


Figura 4: Representación esquemática de posibles soluciones. Se evidencia que el sistema (2.3.1) tiene dos soluciones en el caso general.

2.4 - Construcción de una elipse tangente a otras dos.

Inicialmente es necesario explicar cómo hallar el centro c de una circunferencia de radio R tangente a dos curvas $g_i(t)$, $i = \overline{1,2}$.

La tangencia entre g_i y la circunferencia implica que c se encuentra sobre la curva

$h_i = g_i + RN_i$, donde N_i es una de las normales unitarias $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} g'_i / |g'_i|$. Luego, c es la intersección de las dos curvas h_i , o la expresión $h_j(t_0)$ para cualquier $j \in \{1, 2\}$, donde t_0 es la solución de la ecuación $h_1(t) = h_2(t)$.

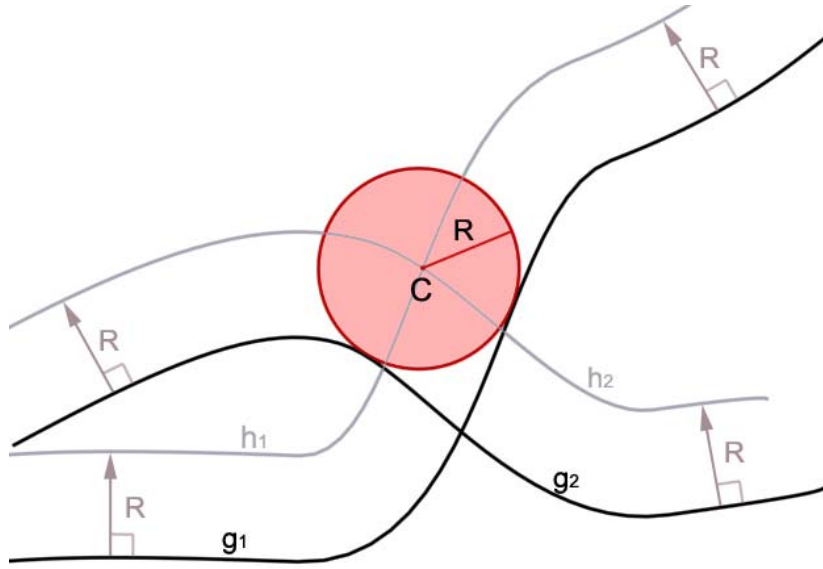


Figura 5: Construcción de una circunferencia tangente a dos curvas.

Se tienen ahora tres elipses de ecuación paramétrica

$$r_i(t) = M_{\theta_i} \begin{pmatrix} a_i \cos t \\ b_i \sin t \end{pmatrix} + c_i, \quad 0 \leq t \leq 2\pi, \quad i = \overline{1,3},$$

donde M_{θ_i} es la transformación que aplicada a un punto lo rota un ángulo θ_i alrededor del origen de coordenadas. Se desea conocer las coordenadas del punto c_3 sabiendo que el elipsoide r_3 es tangente a los r_i , $i = \overline{1,2}$.

Si se le aplica la transformación biyectiva

$$T_R : \mathbb{R}^2 \rightarrow \mathbb{R}^2$$

$$x \mapsto R \begin{pmatrix} 1/a_3 & 0 \\ 0 & 1/b_3 \end{pmatrix} M_{-\theta_3} x,$$

a todo el espacio, se obtiene que $T_R r_3$ es una circunferencia de centro $T_R c_3$ y radio R tangente a las curvas $T_R r_i$, $i = \overline{1,2}$. Luego, la expresión

$$T^{-1} \tilde{c}_3 = \frac{1}{R} M_{\theta_3} \begin{pmatrix} a_3 & 0 \\ 0 & b_3 \end{pmatrix} \tilde{c}_3,$$

es el centro de la elipse r_3 tangente a las otras dos.

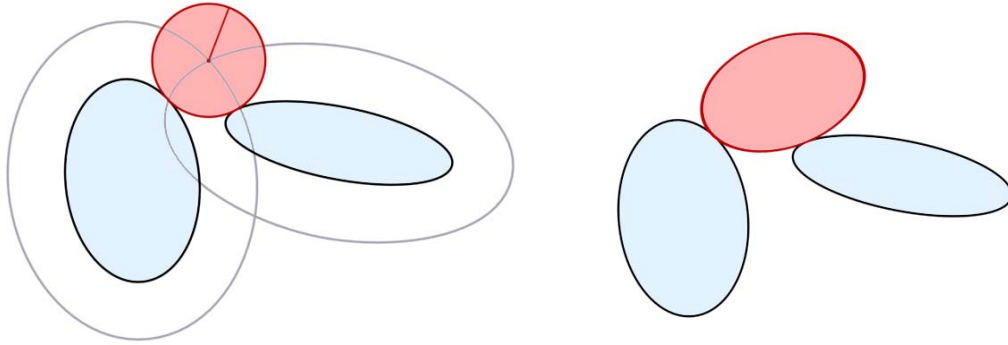


Figura 6: Elipse tangente a otras dos.

2.5 - Construcción de un elipsoide tangente a otros tres.

Inicialmente es necesario explicar cómo hallar el centro c de una esfera de radio R tangente a tres superficies $g_i(u, v)$, $i = \overline{1, 3}$.

La tangencia entre g_i y la esfera implica que c se encuentra sobre la superficie $h_i = g_i + RN_i$, donde N_i es una de las normales unitarias $\pm \frac{\partial g_i / g_u \times \partial g_i / g_v}{|\partial g_i / g_u \times \partial g_i / g_v|}$. Luego, c es la intersección de las

tres superficies h_i , o la expresión $h_j(u_0, v_0)$ para cualquier $j \in \{1, 2, 3\}$, donde (u_0, v_0) es la solución del sistema de ecuaciones

$$\begin{cases} h_1(u, v) - h_2(u, v) = 0 \\ h_1(u, v) - h_3(u, v) = 0 \end{cases}$$

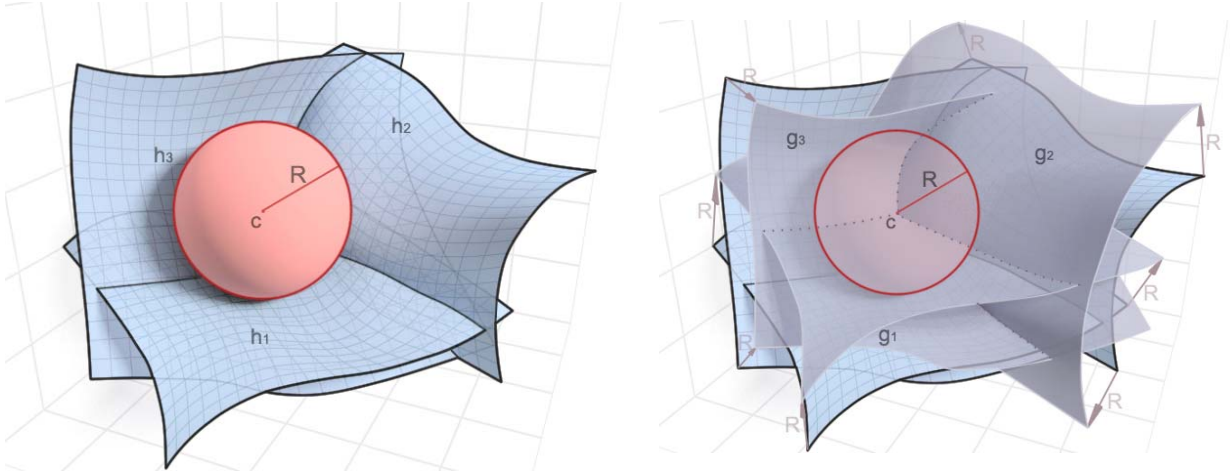


Figura 7: Construcción de una esfera tangente a tres superficies.

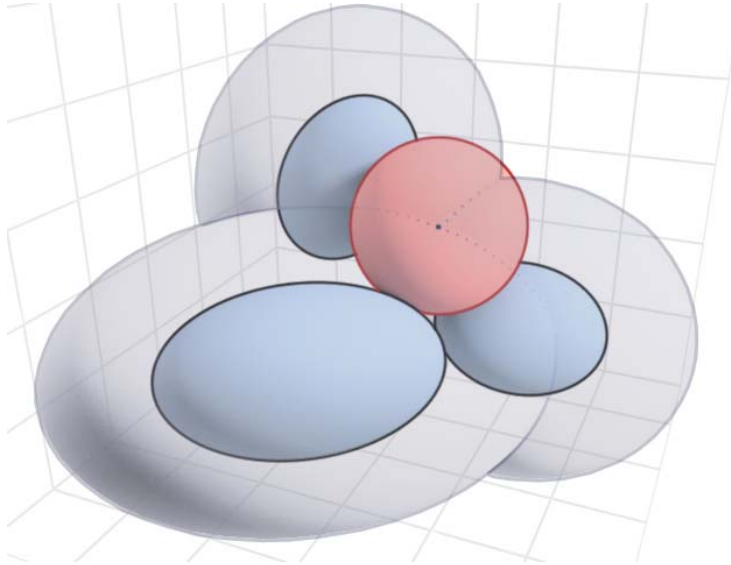


Figura 8: Construcción de una esfera tangente a tres elipsoides.

Se tienen ahora cuatro elipsoides de ecuación paramétrica

$$r_i(u, v) = M_{\theta_{i1}, \theta_{i2}, \theta_{i3}} \begin{pmatrix} a_{i1} \sin v \cos u \\ a_{i2} \sin v \sin u \\ a_{i3} \cos v \end{pmatrix} + c_i, \\ 0 \leq u \leq 2\pi, \quad 0 \leq v < \pi, \quad i = \overline{1, 4},$$

donde $M_{\theta_{i1}, \theta_{i2}, \theta_{i3}}$ es la transformación que aplicada a un punto lo rota un ángulo θ_{i1} alrededor del eje x , luego un ángulo θ_{i2} alrededor del eje y y por último un ángulo θ_{i3} alrededor del eje z .

Se desea conocer las coordenadas del punto c_4 sabiendo que el elipsoide r_4 es tangente a los r_i , $i = \overline{1,3}$.

Si se le aplica la transformación biyectiva

$$T_R : R^3 \rightarrow R^3$$

$$x \mapsto R \begin{pmatrix} 1/a_{41} & 0 & 0 \\ 0 & 1/a_{42} & 0 \\ 0 & 0 & 1/a_{43} \end{pmatrix} M_{-\theta_{43}, -\theta_{42}, -\theta_{41}} x$$

a todo el espacio, se obtiene que $T_R r_4$ es una esfera de centro $T_R c_4$ y radio R tangente a las curvas $T_R r_i$, $i = \overline{1,3}$. El centro \tilde{c}_4 de dicha esfera es posible hallarlo usando lo explicado al inicio del epígrafe. Luego, la expresión es el centro del elipsoide r_4 tangente a los otros tres.

$$T^{-1} \tilde{c}_4 = \frac{1}{R} M_{\theta_{41}, \theta_{42}, \theta_{43}} \begin{pmatrix} a_{41} & 0 & 0 \\ 0 & a_{42} & 0 \\ 0 & 0 & a_{43} \end{pmatrix} \tilde{c}_4$$

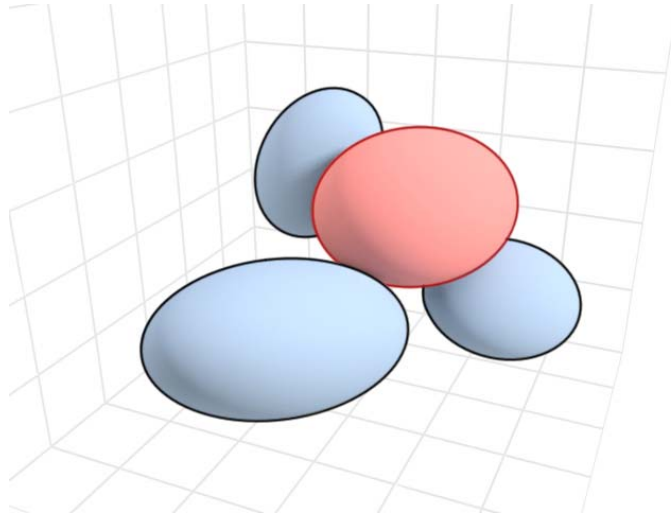


Figura 9: Elipsoide tangente a otros tres.

2.6 - Aproximación de elipses y elipsoides de revolución mediante circunferencias y esferas respectivamente.

Debido a la complejidad que puede presentar la implementación de lo expuesto en los dos epígrafes anteriores, es más conveniente aproximar las elipses mediante circunferencias y esferas respectivamente y usar entonces lo expuesto en los dos epígrafes siguientes.

Para aproximar una elipse, una vía conveniente consiste en colocar circunferencias a lo largo del segmento que une los focos con un radio tal que la circunferencia sea tangente a la elipse.

Para aproximar un elipsoide de revolución de la misma manera, basta considerar esferas con el mismo centro y radio que las circunferencias que aproximan la elipse que al rotar sobre su eje mayor describe al elipsoide.

Considérese una elipse de ecuación $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$, $a > b > 0$. Si una circunferencia tiene su centro en un punto $(l, 0)$ del segmento que une los focos, su radio debe ser igual a la solución del siguiente problema de optimización para lograr la tangencia entre las curvas:

$$\begin{aligned} \min \quad & (x-l)^2 + y^2 \\ \text{s.a.} \quad & \frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \end{aligned}$$

Este problema tiene solución inmediata.

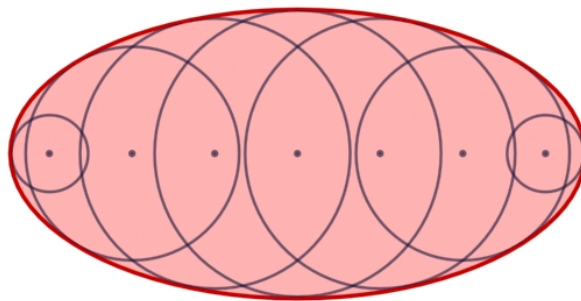


Figura 10: Elipse aproximada con siete circunferencias.

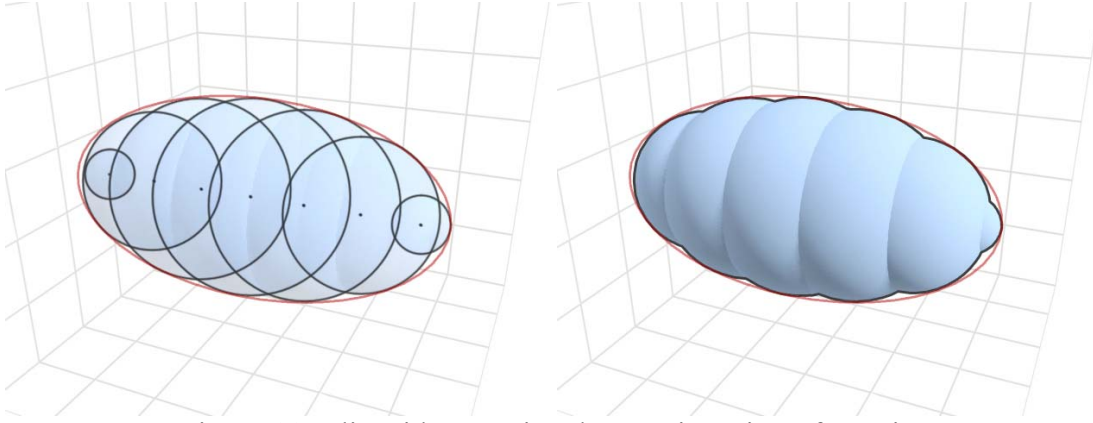


Figura 11: Elipsoide aproximado con siete circunferencias.

2.7 - Construcción de una partícula formada por colisión de circunferencias tangente a otras dos(caso 2D).

Para una mejor comprensión del problema, es conveniente primeramente comenzar en dos dimensiones para luego extenderlo al espacio tridimensional. Por lo tanto, considérese inicialmente el problema de construir una partícula formada por colisión de círculos(p.f.c.c.) tangente a otras dos, el cual se enuncia a continuación:

Se representará a cada p.f.c.c. mediante el elemento $(c_1, r_1, \dots, c_n, r_n)$, siendo c_i el centro de cada círculo y r_i su radio respectivamente, para $i = \overline{1, n}$. Sean $p_i = (c_{i1}, r_{i1}, \dots, c_{in_i}, r_{in_i})$, $i = \overline{1, 3}$, tres p.f.c.c. Considérese el problema de trasladar a p_3 de manera que sea tangente exteriormente a p_1 y p_2 sin cortarlas.

Solución:

El problema equivale a hallar un vector c tal que la partícula $p'_3 = (c'_{31}, r'_{31}, \dots, c'_{3n_3}, r'_{3n_3}) = (c_{31} + c, r_{31}, \dots, c_{3n_3} + c, r_{3n_3})$ sea tangente exteriormente a p_1 y p_2 sin cortarlas.

Primeramente se verá cómo construir un conjunto finito A que contiene todos los posibles valores de c que hacen tangente a p'_3 con p_1 y p_2 . Para descartar de A los valores de c que provocan que p'_3 corte a p_1 o a p_2 , se descartan todas las partículas p'_3 que tengan algún círculo componente que corte a algún círculo componente de p_1 o de p_2 .

Las siguientes equivalencias son fácilmente demostrables:

$$\begin{aligned}
 & p_1 \text{ es tangente a } p'_3 \\
 & \Leftrightarrow \exists i, j : C(c_{1i}, r_{1i}) \text{ es tangente a } C(c'_{3j}, r_{3j}) \\
 & \Leftrightarrow \exists i, j : c'_{3j} \in C(c_{1i}, r_{1i} + r_{3j}) \\
 & \Leftrightarrow \exists i, j : c_{3j} + c \in C(c_{1i}, r_{1i} + r_{3j}) \\
 & \Leftrightarrow \exists i, j : c \in C(c_{1i} - c_{3j}, r_{1i} + r_{3j}) \\
 & \Leftrightarrow c \in A_1 = \bigcup_{i,j} C(c_{1i} - c_{3j}, r_{1i} + r_{3j}).
 \end{aligned}$$

Siguiendo pasos análogos se puede demostrar que

$$c \in A_2 = \bigcup_{i,j} C(c_{2i} - c_{3j}, r_{2i} + r_{3j}),$$

de donde $c \in A = A_1 \cap A_2$. Esta última intersección se halla intersectando cada una de las circunferencias que componen A_1 , con cada una de las que compone A_2 . Por último, para descartar los valores no admisibles de c , se desechan todas las partículas p'_3 que tengan algún círculo componente que corte a alguno de los que componen a p_1 o p_2 .

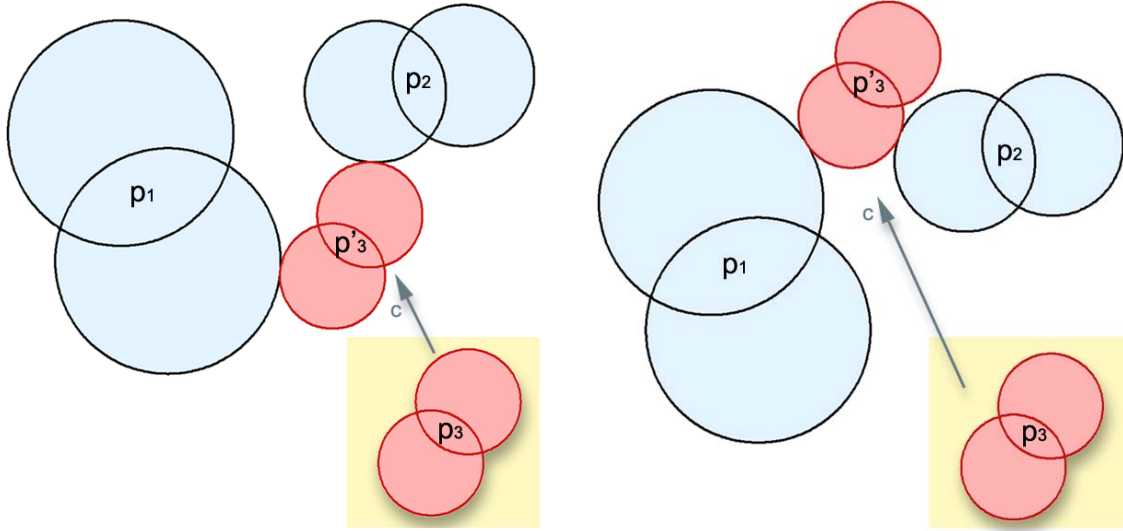


Figura 12: Construcción de una p.f.c.c. tangente a otras dos.

La partícula roja es trasladada para que sea tangente a las otras dos. Nótese que en el primer caso hay tangencia en circunferencias distintas de la partícula roja.

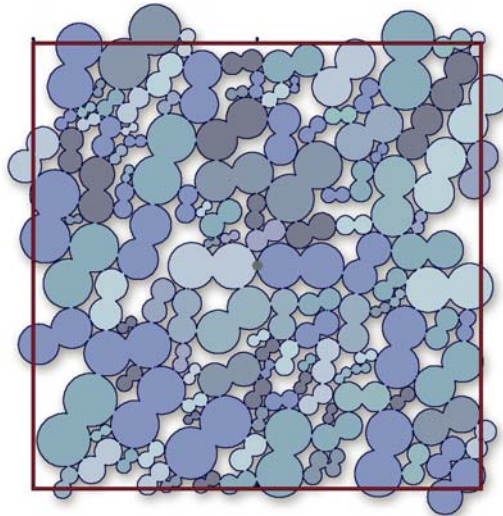


Figura 13: Empaquetamiento de partículas formadas por colisión de dos círculos.

2.8 - Construcción de una partícula formada por colisión de esferas tangente a otras tres(caso 3D).

Análogamente al epígrafe anterior, en tres dimensiones el problema es el siguiente:

Se representará a cada partícula formada por colisión de esferas(p.f.c.e.) mediante el elemento $(c_1, r_1, \dots, c_n, r_n)$, siendo c_i el centro de cada esfera y r_i su radio respectivamente, para $i = \overline{1, n}$. Sean $p_i = (c_{i1}, r_{i1}, \dots, c_{in_i}, r_{in_i})$, $i = \overline{1, 4}$, cuatro p.f.c.e. Trasládese a p_4 de manera que sea tangente exteriormente a p_1, p_2 y p_3 sin cortarlas.

Solución:

El problema equivale a hallar un vector c tal que la partícula $p'_4 = (c'_{41}, r'_{41}, \dots, c'_{4n_4}, r'_{4n_4}) = (c_{41} + c, r_{41}, \dots, c_{4n_4} + c, r_{4n_4})$. Como la deducción es prácticamente la misma que en el caso bidimensional, solamente se escribirá el resultado final.

Se tiene que

$$c \in A_k = \bigcup_{i,j} E(c_{ki} - c_{4j}, r_{ki} + r_{4j}), \quad k = \overline{1, 3}.$$

Luego, para obtener todos los posibles valores de c es necesario hallar todos los puntos de intersección de tríos de esferas, estando cada una incluida en un A_k distinto. Por último, para descartar los valores no admisibles de c , se desechan todas las partículas p'_4 que tengan alguna esfera componente que corte a alguna de las que componen a p_1, p_2 o p_3 .

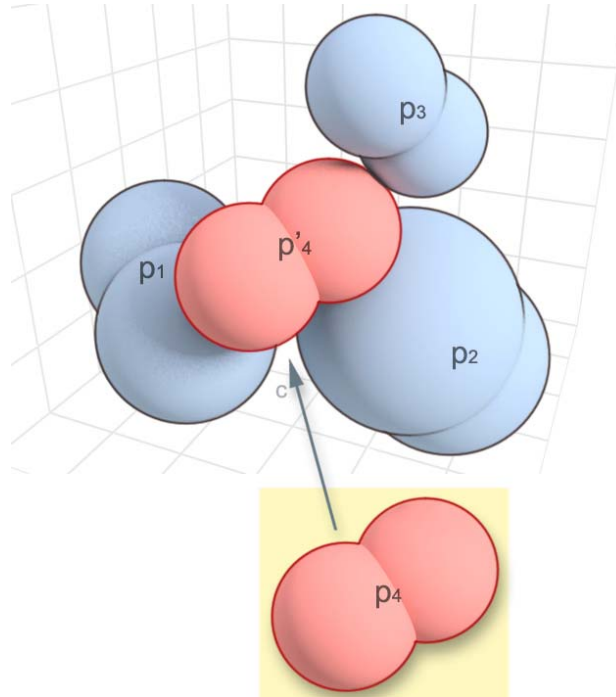


Figura 15: Construcción de una p.f.c.e. tangente a otras tres.

2.9 – Construcción de un polígono que está en contacto con otros dos.

Sean $\tilde{P}_i, i = \overline{1,3}$, tres polígonos convexos de centro de masas C_i y vértices $V_{ij}, j = \overline{1, n_i}$, con $r_{ij} = V_{ij} - C_i$. Considérese el problema de hallar C_3 de manera que \tilde{P}_3 esté en contacto con \tilde{P}_1 y \tilde{P}_2 sin cortarlos.

La idea de la solución es hallar primeramente el lugar geométrico descrito por C_3 al deslizar cada vértice de \tilde{P}_3 por cada lado de \tilde{P}_1 y al deslizar cada lado de \tilde{P}_3 por cada vértice de \tilde{P}_1 . Sea \tilde{A}_1 este lugar geométrico, el cual es una unión finita de segmentos.

Si se construye \tilde{A}_2 de manera análoga sustituyendo \tilde{P}_1 por \tilde{P}_2 , se tiene que $\tilde{C}_3 \in \tilde{A}_1 \cap \tilde{A}_2$. O sea, todos los posibles valores de C_3 se obtienen intersectando cada segmento contenido en \tilde{A}_1 con cada segmento contenido en \tilde{A}_2 .

A continuación se muestra cómo construir \tilde{A}_1 , ya que \tilde{A}_2 se obtiene de una manera análoga.

Los dos casos posibles de contacto entre \tilde{P}_1 y \tilde{P}_3 son los siguientes(ver figuras):

- i) (vértice de \tilde{P}_1) – (lado de \tilde{P}_3)
- ii) (lado de \tilde{P}_1) – (vértice de \tilde{P}_3)

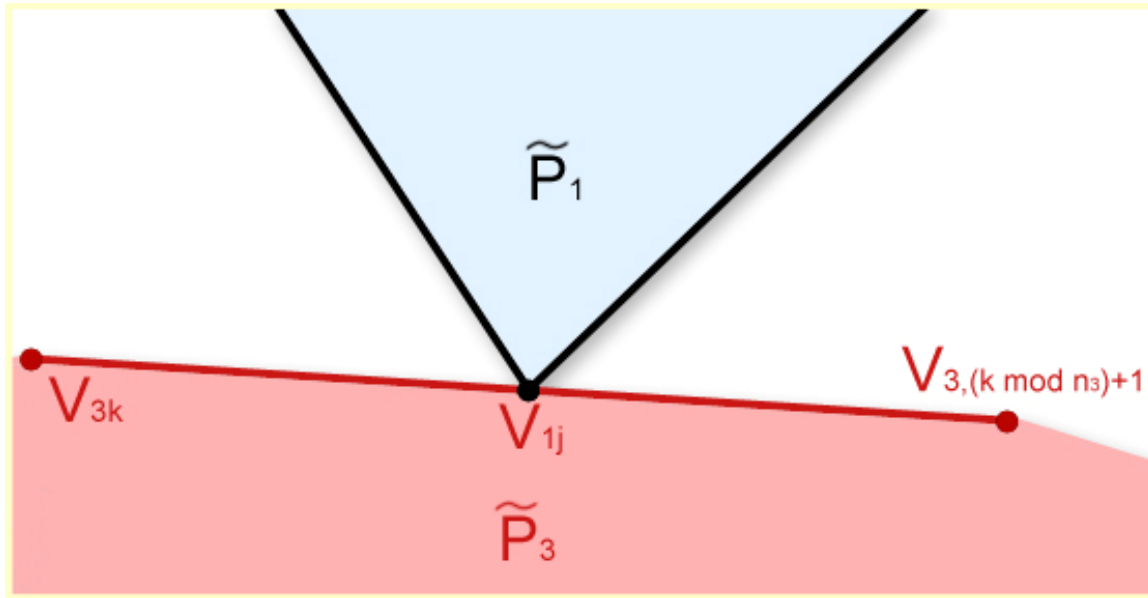


Figura 16: Caso de contacto. Caso i)

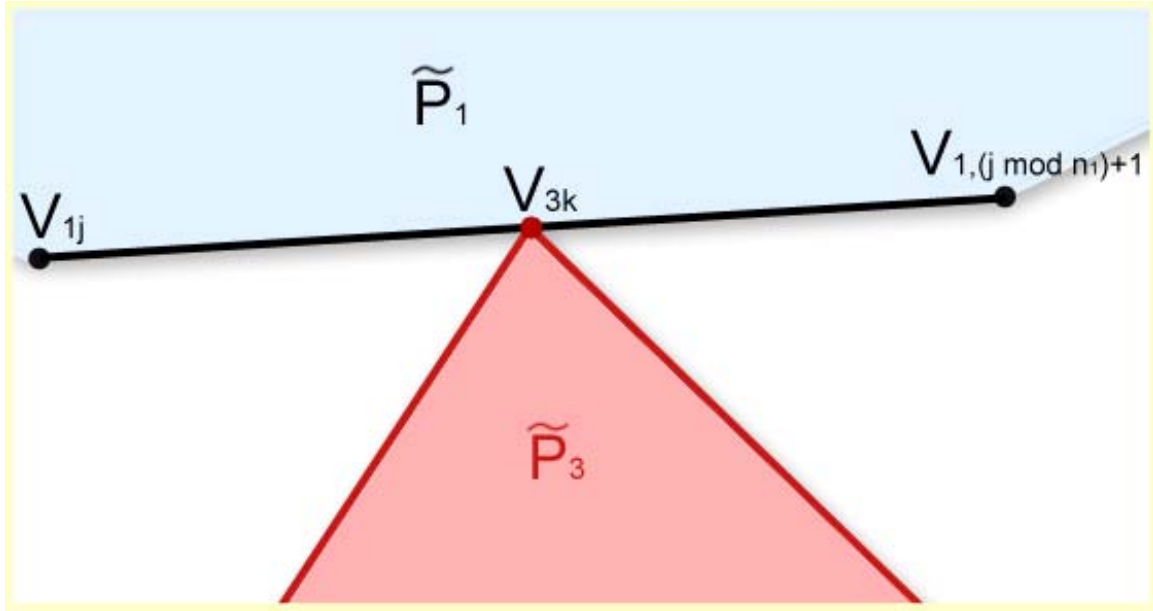


Figura 17: Caso de Contacto. Caso ii)

Para el caso i), se debe agregar a \tilde{A}_1 el segmento $s_1 = \overline{V_{3k} - r_{3k}}, \overline{V_{3k_0} - r_{3k_0}}, k = \overline{1, n_3}, k_0 = (k \bmod n_3) + 1$, si se cumple que C_1 y $V_{1j} - r_{3k}$ están a distintos lados de la recta que contiene a s_1 .

Para el caso ii), se debe agregar a \tilde{A}_1 el segmento $s_2 = \overline{V_{1j} - r_{3k}}, \overline{V_{1j_0} - r_{3k}}, j = \overline{1, n_1}, j_0 = (j \bmod n_1) + 1$, si se cumple que C_1 y $V_{1j} - r_{3k}$ están a distintos lados de la recta que contiene a s_2 .

Por último, se deben descartar los valores de C_3 para los cuales \tilde{P}_3 corta a \tilde{P}_1 o a \tilde{P}_2 . Para saber cómo se determina si dos polígonos se cortan, véase el epígrafe 2.1.

Las siguientes figuras (figura 15 y 16) muestran ejemplos de empaquetamiento de cuadrados y polígonos respectivamente, logrados mediante la utilización de lo explicado en este epígrafe.

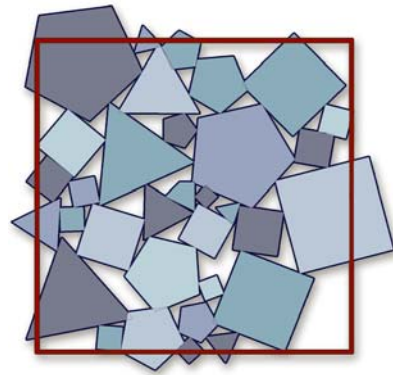
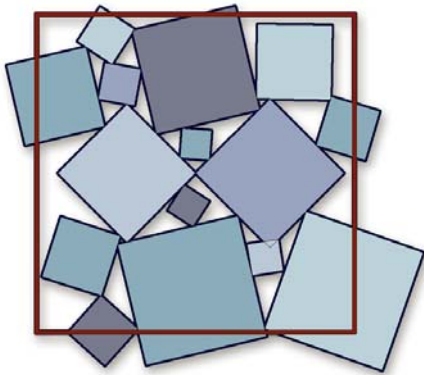


Figura 18: Empaquetamiento de cuadrados Figura 19: Empaquetamiento de polígonos

2.10 - Construcción de un poliedro que está en contacto con otros tres.

Sean $\tilde{P}_i, i = \overline{1,4}$, cuatro poliedros convexos de centro de masas C_i , vértices $V_{ij}, j = \overline{1, n_i}$, y caras $C_{i, p_1 p_2 p_3}$ de vértices V_{p_1}, V_{p_2} y V_{p_3} , donde se asume que las caras son triángulos para mayor comprensión del problema. Sean además $r_{ij} = V_{ij} - C_i$ y $d_{ik_1 k_2} = V_{ik_1} - V_{ik_2}$. Considérese el problema de hallar C_4 de manera que \tilde{P}_4 esté en contacto con \tilde{P}_1, \tilde{P}_2 y \tilde{P}_3 sin cortarlos.

La idea de la solución es hallar primeramente el lugar geométrico descrito por C_4 al deslizar cada vértice, cara y lado de \tilde{P}_4 por cada cara, vértice y lado de \tilde{P}_1 respectivamente. Sea \tilde{A}_1 este lugar geométrico, el cual es una unión finita de triángulos y paralelogramos en el espacio.

Si se construyen de manera análoga \tilde{A}_2 y \tilde{A}_3 sustituyendo \tilde{P}_1 por \tilde{P}_2 y \tilde{P}_3 respectivamente, se tiene que $\tilde{C}_4 \in \tilde{A}_1 \cap \tilde{A}_2 \cap \tilde{A}_3$. O sea, C_4 se encuentra en el conjunto de las intersecciones de los elementos de los tríos que se forman al tomar un polígono incluido en \tilde{A}_1 , otro en \tilde{A}_2 y otro en \tilde{A}_3 .

Los tres casos posibles de contacto entre \tilde{P}_1 y \tilde{P}_4 son los siguientes (figuras 17, 18 y 19):

- i) (vértice de \tilde{P}_1) – (cara de \tilde{P}_4)
- ii) (cara de \tilde{P}_1) – (vértice de \tilde{P}_4)
- ii) (lado de \tilde{P}_1) – (lado de \tilde{P}_4)

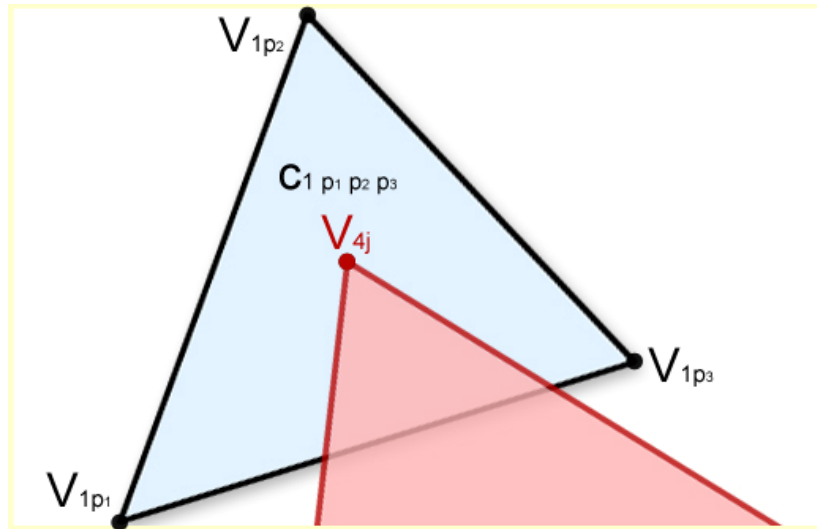


Figura 20: Caso de Contacto. Caso i)

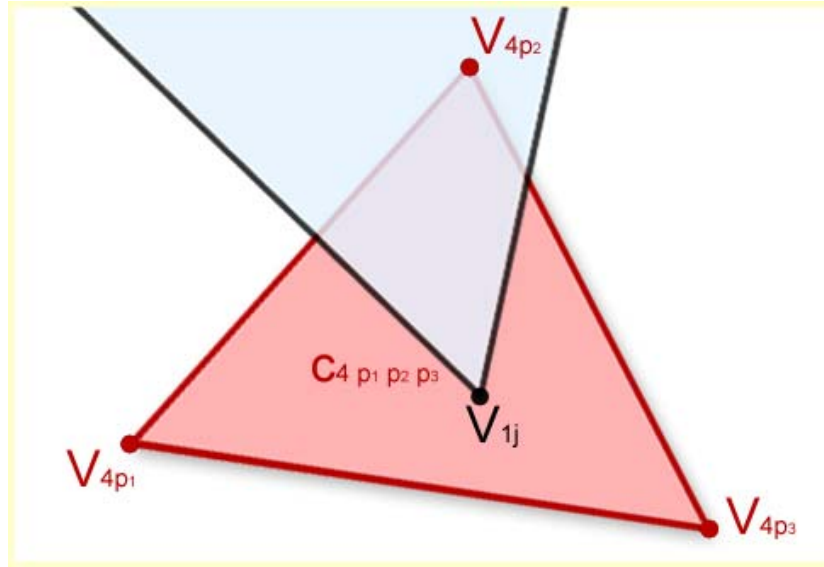


Figura 21: Caso de Contacto. Caso ii)

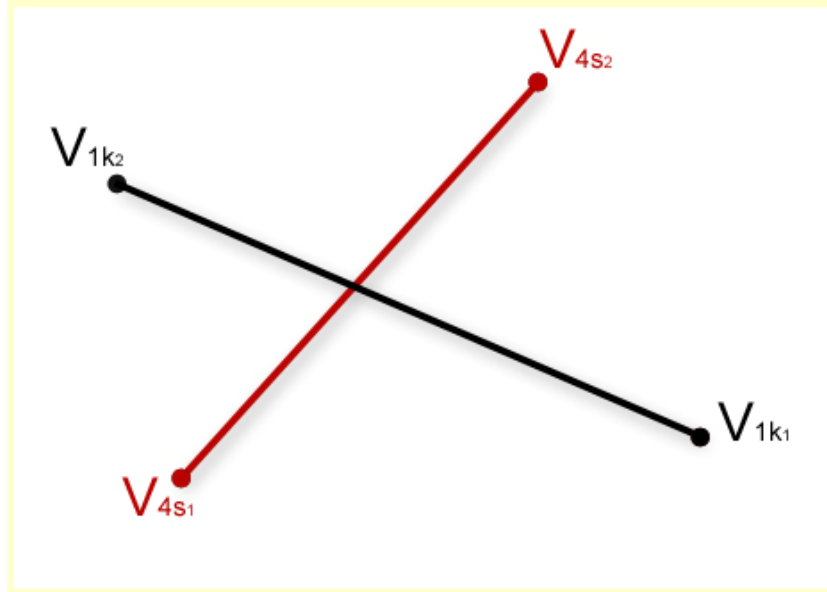


Figura 22: Caso de Contacto. Caso iii)

Para el caso i), se debe agregar a \tilde{A}_1 el triángulo de vértices $V_{1p_i}-r_{4j}$, $i = \overline{1,3}$, si se cumple que C_1 y $V_{1p_i}-r_{4j}$ están a distintos lados del plano que contiene al $\Delta V_{1p_1}V_{1p_2}V_{1p_3}$.

Para el caso ii), se debe agregar a \tilde{A}_1 el triángulo de vértices $V_{1j}-r_{4p_i}$, $i = \overline{1,3}$, si se cumple que C_1 y $V_{1j}-r_{4p_i}$ están a distintos lados del plano que pasa por V_{1j} y es paralelo al $\Delta r_{4p_1}r_{4p_2}r_{4p_3}$.

Para el caso iii), se debe agregar a \tilde{A}_1 el paralelogramo de vértices $V_{1k_i} \pm (r_{4s_2} - r_{4s_1})$, $i = \overline{1,2}$, si se cumple que $(r_{4s_2} - r_{4s_1}) \times (r_{1k_2} - r_{1k_1}) \neq 0$; y C_1 y $V_{1k_1} - r_{4s_1}$ están a distintos lados del plano que contiene a V_{1k_1} y es paralelo al $\Delta V_{1k_1}(V_{1k_1} + r_{4s_2} - r_{4s_1})V_{1k_2}$.

Por último, se deben descartar los valores de C_4 para los cuales \tilde{P}_4 corta a \tilde{P}_1 , \tilde{P}_2 o a \tilde{P}_3 . Para saber cómo se determina si dos poliedros se cortan, véase el epígrafe 2.1.

2.11 - Método de interioridad en dos dimensiones.

Cuando se está generando el medio físico, es necesario garantizar que cada partícula esté ubicada dentro del conjunto de segmentos (contorno) que se quiere llenar. El método creado por el autor de este trabajo y descrito a continuación tiene por objetivo determinar cuándo un punto se encuentra dentro de un conjunto cualquiera de segmentos (que define el contorno).

Considérese que se tiene un punto dado P y un conjunto de segmentos. La idea general a seguir consiste en buscar el segmento del conjunto de segmentos más cercano a P y luego, conociendo una normal relativa al segmento que apunte hacia fuera del conjunto de segmentos, es posible determinar si P está dentro o fuera de este.

Se puede ver que todos los puntos que están en un mismo semiplano determinado por un lado del conjunto de segmentos y que están suficientemente próximos a dicho lado, cumplen la propiedad de estar todos dentro o todos fuera.

Un método comúnmente usado para resolver este mismo problema consiste en trazar un rayo con origen en P y contar las intersecciones de dicho rayo con el conjunto de segmentos. Esta vía tiene la desventaja de que puede producir resultados incorrectos si no se escoge el rayo convenientemente, por ejemplo, cuando este pasa por el extremo de algún lado del conjunto de segmentos.

Imagínese una circunferencia con centro en P, cuyo radio va aumentando desde cero hasta tocar un punto de algún segmento s_0 del conjunto de segmentos (al que se denotará por M). Si el contacto entre la esfera y s_0 ocurre en un punto del interior de este último, se tiene que P es exterior a M si y solo si P se encuentra en el semiplano (determinado por la recta que contiene a s_0) hacia el cual apunta la normal a s_0 exterior a M.

Supóngase ahora que el contacto entre la circunferencia y s_0 ocurre en un extremo p_0 de un lado de M. En ese caso, P es exterior respecto a M si y solo si el ángulo no sobreobtusos con vértice en p_0 es “interior” a M. Si los lados de M están orientados de manera que hacia la izquierda de cualquier lado está el interior de M y a la derecha se encuentra el exterior, es fácil determinar si el ángulo no sobreobtusos con vértice en p_0 es “interior” a M. Al vértice p_0 se le llamará “saliente” si y solo si el mencionado ángulo es “interior” a M.

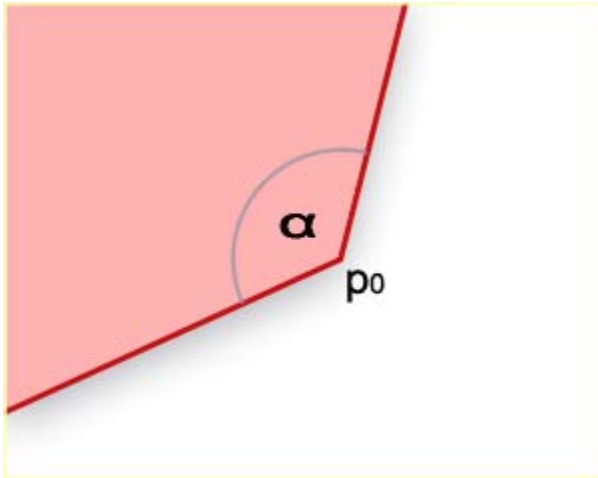


Figura 23: Ángulo no sobreobtuso α con vértice en p_0 saliente.

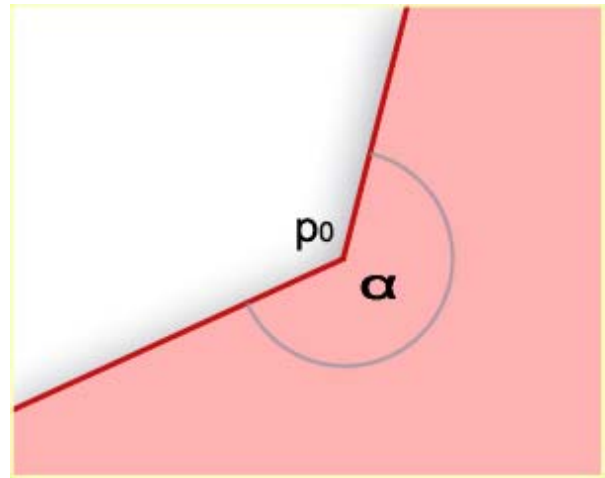


Figura 24: Ángulo no sobreobtuso α con vértice en p_0 entrante.

En la siguiente figura 22, se ha construido una malla y mediante el método de interioridad anteriormente descrito, los puntos interiores al conjunto de segmentos se han puesto de color rojo y los exteriores de color gris.

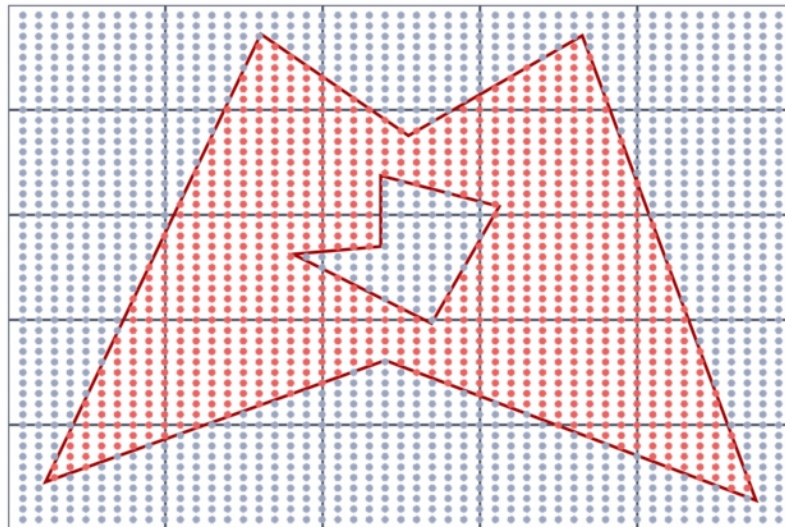


Figura 25: Malla donde los puntos interiores al conjunto de segmentos están en rojo y los exteriores en gris.

En Berg [2] es posible encontrar un algoritmo mucho más eficiente, aunque no fácilmente extensible a tres dimensiones. En este caso la formulación realizada presenta la novedad de ser un algoritmo fiable, eficiente y robusto en 2D y 3D.

2.12 - Método de interioridad en tres dimensiones.

Cuando se está generando el medio físico, es necesario garantizar que cada partícula esté ubicada dentro de la malla (que define el contorno en 3D) que se quiere llenar. El método elaborado en este trabajo y descrito a continuación, tiene por objetivo determinar cuándo un punto se encuentra dentro de una malla cualquiera de triángulos.

Considérese que se tiene un punto dado P y una malla de triángulos. La idea general a seguir consiste en buscar el triángulo de la malla más cercano a P y luego, conociendo una normal relativa al triángulo que apunte hacia fuera de la malla, es posible determinar si P está dentro o fuera de esta.

Se puede ver que todos los puntos que están en un mismo semiespacio determinado por un triángulo de la malla y que están suficientemente próximos a dicho triángulo, cumplen la propiedad de estar todos dentro o todos fuera.

Un método comúnmente usado para resolver este mismo problema consiste en trazar un rayo con origen en P y contar las intersecciones de dicho rayo con la malla. Esta vía tiene la desventaja de que puede producir resultados incorrectos si no se escoge el rayo convenientemente, por ejemplo, cuando este pasa por un lado de algún triángulo de la malla.

Imagínese una esfera con centro en P , cuyo radio va aumentando desde cero hasta tocar un punto de algún triángulo Δ_0 de la malla (a la que se denotará por M). Si el contacto entre la esfera y Δ_0 ocurre en un punto del interior de este último, se tiene que P es exterior a M si y solo si P se encuentra en el semiespacio (determinado por el plano que contiene a Δ_0) hacia el cual apunta la normal a Δ_0 exterior a M .

Supóngase ahora que el contacto entre la esfera y Δ_0 ocurre en un punto de un lado l de Δ_0 distinto de los extremos. En ese caso, P es exterior respecto a M si y solo si el ángulo no sobreobtusos formado por los triángulos que comparten l es “interior” a M . Para determinar si un ángulo no sobreobtusos formado por dos triángulos es interior a M se procede de la siguiente manera:

Sean $\Delta_1, \Delta_2 \in M$, donde los vértices de Δ_1 son T, U y V y los de Δ_2 son T, U y W . Obviamente, Δ_1 y Δ_2 comparten el lado \overline{TU} . Entonces, el ángulo no sobreobtusos formado por Δ_1 y Δ_2 es interior a M si y solo si la normal exterior relativa a Δ_2 apunta hacia el semiespacio (determinado por el plano que contiene a Δ_2) que no contiene a V . Si esto último ocurre, al lado \overline{TU} se le llamará saliente.

Finalmente, supóngase que el punto en el que la esfera toca a Δ_0 es en un vértice de este. Entonces P es exterior a M si y solo si el “pico” de vértice V es “interior” a M . Para determinar si un pico con vértice en V es interior se procede de la siguiente manera:

Sean v_1, \dots, v_n los vectores unitarios con origen en V cuya dirección coincide con la de los lados de los triángulos de M que tienen a V como uno de sus extremos. Sean n la normal exterior a un triángulo que contiene a V y v_0 definida por:

$$v_0 = \underset{s.a.}{\operatorname{argmin}} \max_{\substack{1 \leq i \leq n \\ \|v\|=1}} v \cdot v_i.$$

Entonces, el pico de vértice V es interior a M si y solo si $v_0 \cdot n \geq 0$. Si esto ocurre, al vértice V de la malla se le llamará “saliente”.

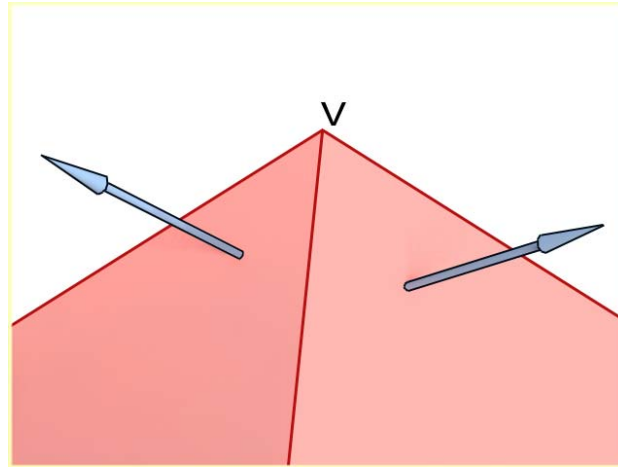


Figura 26: Vértice saliente de la malla.

Todo lo anterior conduce al siguiente algoritmo para determinar si un punto es interior a una malla de triángulos:

Algoritmo de Interioridad(P).

Entrada: punto P

Salida: dentro (esta variable puede tomar los valores Verdadero o Falso)

1- Buscar el triángulo Δ_0 de M más cercano a P. Sean V_1, V_2 y V_3 sus vértices.

-Sea Q el punto más cercano a P tal que $Q \in \Delta_0$.

2- Si $Q \notin \overline{V_1V_2} \cup \overline{V_1V_3} \cup \overline{V_2V_3}$, entonces

2.1- dentro $\leftarrow (P - V_1) \cdot n \leq 0$, donde n es una normal exterior a Δ_0

2.2- terminar

3- Si $Q \in \overline{V_iV_j}$ para algunos i, j , entonces

3.1- si el lado $\overline{V_iV_j}$ es saliente, entonces

dentro \leftarrow Falso

de lo contrario

dentro \leftarrow Verdadero

3.2- terminar

- 4- Si $Q = V_i$ para algún i , entonces
 - 4.1 si V_i es saliente entonces
 - dentro \leftarrow Falso
 - de lo contrario
 - dentro \leftarrow Verdadero
 - 4.2- terminar

Notas:

- 1) En el algoritmo anterior no es necesario conocer Q explícitamente.
- 2) En el paso 1, la búsqueda del triángulo más cercano a P se debe hacer usando la menor cantidad posible de triángulos de la malla. Una vía factible es encerrar la malla en una caja acotadora y particionar esta en celdas contiguas en forma de ortoedro. Luego, para cada una de estas celdas C , se asignan los triángulos de M que cumplan la propiedad de ser el más cercano a algún punto $P_i \in C$. Finalmente, se puede asegurar que si P se encuentra en una celda determinada, el triángulo más cercano a él es uno de los previamente asignados a esa celda.

Una segunda vía para esto, que implica grandes dificultades, es construir un diagrama de Voronoi con la distancia euclidiana usual, donde los sitios son los triángulos que componen M .

2.13 - Algoritmo de empaquetamiento de partículas (en tres dimensiones).

Se presenta aquí un algoritmo para llenar con partículas definidas por parámetros aleatorios, un cuerpo dado. La idea general es bien simple: consiste en tomar una partícula p_0 cualquiera de las que ya se tienen, que pertenezca a la “capa exterior” del conjunto de partículas. Luego, se buscan dos partículas p_1 y p_2 cercanas a p_0 , ambas pertenecientes también a la capa exterior, de manera que sea posible construir una cuarta p_3 tangente a las tres anteriores. Entonces, a partir de p_1 , p_2 y p_3 , se rodea de nuevas partículas a p_0 y a esta última se le deja de considerar parte de la capa exterior. Todo esto se repite hasta tener lleno el cuerpo dado.

A continuación aparece el algoritmo creado y al final las aclaraciones necesarias:

Algoritmo de empaquetamiento en tres dimensiones.

Entrada: malla M que delimita el cuerpo a llenar con partículas; definición de los parámetros que determinan las partículas.

Salida: arreglo P de todas las partículas obtenidas

1- Construir tres partículas “aceptables”*(1) p_1 , p_2 y p_3 de manera que sea posible construir una cuarta tangente a ellas tres.

2- $P \leftarrow \{p_1, p_2, p_3\}$; $C_{ext} \leftarrow P$

3- mientras $C_{ext} \neq \emptyset$ hacer

- 3.1- $e_0 \leftarrow$ último elemento de C_{ext}
- 3¹.2- $C_{ext} \leftarrow C_{ext} - \{e_0\}$
- 3.3- $P_{vec} \leftarrow Vecinas(e_0)$ *(2)
- 3.4- ordenar el arreglo P_{vec} de acuerdo a la cercanía de sus elementos a e_0
*(3)
- 4- repetir
 - tomar dos elementos $e_1, e_2 \in P_{vec}$
 - hasta poder construir una partícula “aceptable” e_3 tangente a e_0, e_1 y e_2 ; o
 - verificar todas las posibilidades para e_1 y e_2 .
- 5- si en el paso 4 fue posible construir e_3 , entonces
 - 5.1- agregar e_3 al final de P y de C_{ext}
 - 5.2- $P_{vec} \leftarrow \{e_1, e_2, e_3\}$
 - 5.3- repetir
 - 5.3.1- tomar dos elementos $e_i, e_j \in P_{vec} : e_i \neq e_1, e_2$
 - 5.3.2- si es posible construir una partícula aceptable e_k tangente a e_0, e_i y e_j ,
entonces
 - 5.3.3- agregar e_k al final de P, C_{ext} y P_{vec}
hasta verificar todas las posibilidades para e_i y e_j .

Nota: El desempeño del algoritmo depende en gran medida de la eficiencia de la función *Vecinas*. Para la búsqueda de elementos vecinos, véase Müller [3].

Como también es útil tener un algoritmo de empaquetamiento en dos dimensiones, a continuación aparece uno análogo al tridimensional:

Algoritmo de empaquetamiento en dos dimensiones.

Entrada: malla M que delimita el cuerpo a llenar con partículas; definición de los parámetros que determinan las partículas.

Salida: arreglo P de todas las partículas obtenidas

- 1- Construir dos partículas “aceptables” p_1 , y p_2 de manera que sea posible construir una tercera tangente a ellas dos.

¹ Se considera “aceptable” a toda partícula que no corte a ninguna otra en más de un punto(para esto se debe usar la función *Vecinas*, de la cual se habla en la nota siguiente) y que además tenga su centro en el interior de la malla M .

² La función *Vecinas*(e_0) devuelve las partículas que están próximas a e_0 , definiendo la proximidad convenientemente para poder construir nuevas partículas tangentes a e_0 y a las vecinas de esta. Esta función también se debe usar para determinar si una partícula se solapa con otra en más de un punto, haciendo así la menor cantidad posible de verificaciones.

³ La cercanía entre dos partículas es la distancia entre las superficies que las delimitan.

2- $P \leftarrow \{p_1, p_2\}; C_{ext} \leftarrow P$

3- mientras $C_{ext} \neq \emptyset$ hacer

3.1- $e_0 \leftarrow$ último elemento de C_{ext}

3.2- $C_{ext} \leftarrow C_{ext} - \{e_0\}$

3.3- $P_{vec} \leftarrow Vecinas(e_0)$

3.4- ordenar el arreglo P_{vec} de acuerdo a la cercanía de sus elementos a e_0

4- repetir

tomar un elemento $e_1 \in P_{vec}$

hasta poder construir una partícula “aceptable” e_2 tangente a e_0 y e_1 ; o
verificar todas las posibilidades para e_1 .

5- si en el paso 4 fue posible construir e_2 , entonces

5.1- agregar e_2 al final de P y de C_{ext}

5.2- $P_{vec} \leftarrow \{e_1, e_2\}$

5.3- repetir

5.3.1- tomar un elemento $e_i \in P_{vec} : e_i \neq e_1$

5.3.2- si es posible construir una partícula aceptable e_k tangente
a e_0 y e_i ,
entonces

5.3.3- agregar e_k al final de P , C_{ext} y P_{vec}
hasta verificar todas las posibilidades para e_i .

2.14 - Aplicaciones preliminares en la generación de diferentes geometrías.

Con el objetivo de demostrar la efectividad y la eficiencia del algoritmo de generación de un medio con partículas esféricas con diámetro y posición arbitraria, se han generado cuatro geometrías diferentes. Dentro de las geometrías generadas se han efectuado unas muy elementales y otras más complejas. En la figura 27 se muestra un cubo de $1 \times 1 \times 1$ y se ha generado con esferas de posición y diámetro aleatorio, los cuales siguen una distribución normal. En este caso se escogió una distribución normal por tomar una de las diferentes distribuciones discretas o continuas conocidas, ya que el algoritmo implementado puede emplear cualquiera. En la figura 27 se observa una distribución caótica de las partículas que conforman el medio y además se obtiene un buen empaquetamiento.

En las figuras 28, 29 y 30 se representan generaciones de partículas de diámetro y posición aleatoria de geometría de mayor complejidad. En estos casos los diámetros de las partículas siguen una distribución t de Student. En la figura 28 se observa la generación de una geometría irregular donde se evidencia la efectividad del algoritmo desarrollado e implementado computacionalmente.

La figura 29 corresponde a la generación de un muelle, que por su compleja geometría podía haber presentado problemas en la generación. En este caso se ilustran (figura 29) los resultados

de la generación del medio, denotando la generalidad de la formulación. Por su parte en la figura 30 se ha generado con partículas esféricas un toroide y al igual que el resto de las geometrías se obtienen excelentes resultados.

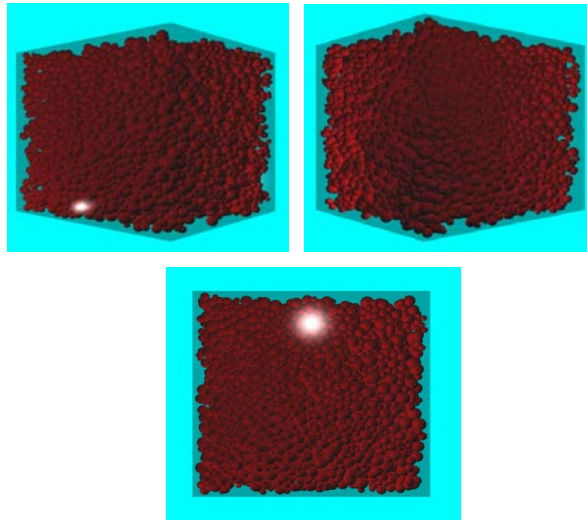


Figura 27: Generación del un Cubo con esferas.

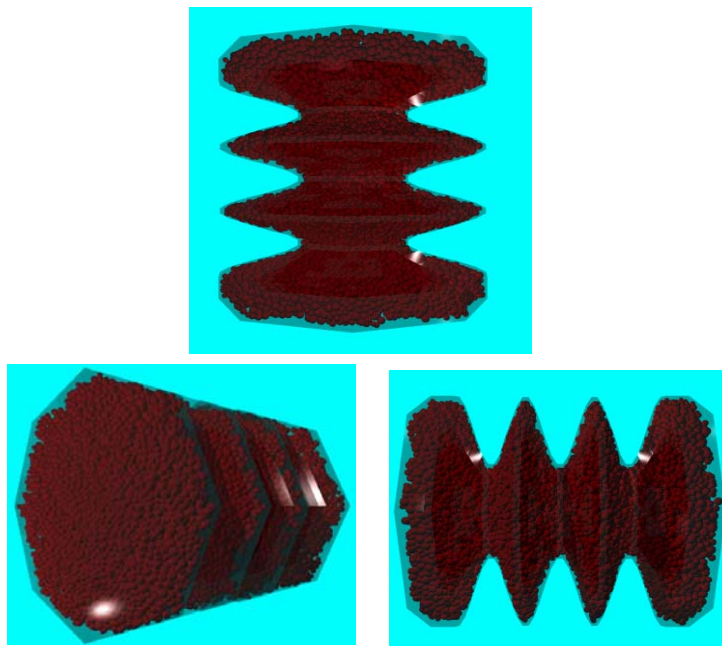


Figura 28: Generación de una geometría en 3D con forma irregular.

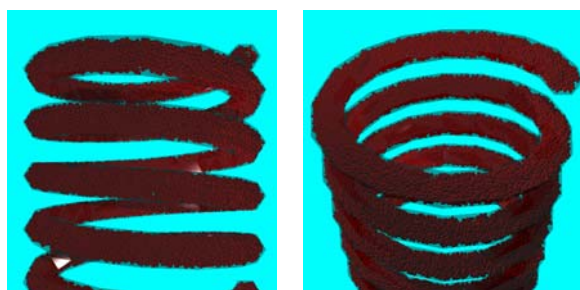


Figura 29: Generación de una geometría en 3D - Muelle.

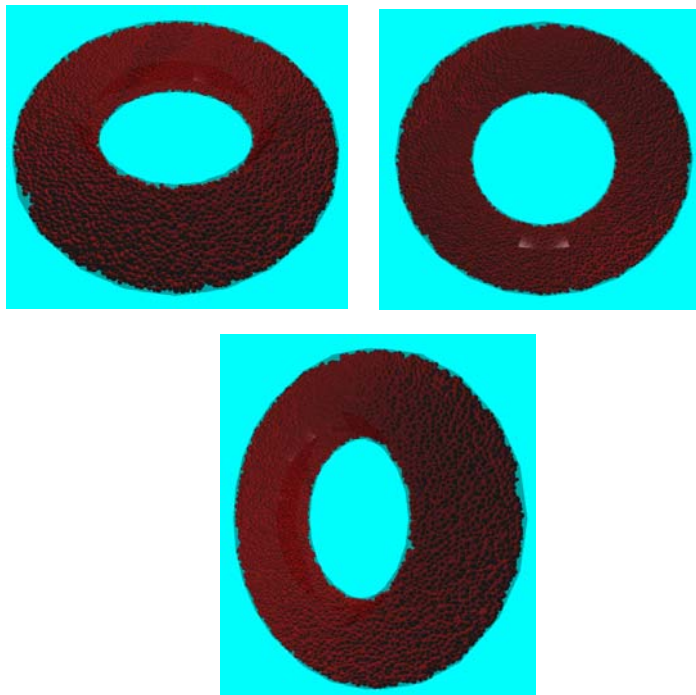


Figura 30: Generación de una geometría en 3D – Toroide.

2.15 – Conclusiones parciales.

En este capítulo se ha desarrollado un método y algoritmo para la generación o empaquetamiento de partículas en un medio (2D y 3D). Para ello se ha desarrollado una serie de aspectos que a modo de conclusión se mencionan a continuación:

- Se presenta una formulación sistémica e integradora que posibilita generar o empaquetar partículas (esferas, colisiones de esferas o clusters de esferas, elipsoides, poliedros, etc.) de diversas geometrías sin necesidad de realizar transformaciones a la formulación original desarrollada. En fin, se ha presentado un algoritmo que permite generar un empaquetamiento independientemente del tipo de partículas con el que se trabaje.
- Se realiza una propuesta de un algoritmo muy eficiente para determinar si un punto se encuentra en el interior de una malla, denominado algoritmo de interioridad. Este algoritmo elimina deficiencias detectadas en otras formulaciones que tratan de determinar estas peculiaridades.
- Se han presentado nuevos métodos numéricos para resolver sistemas de ecuaciones que permiten colocar cuerpos de manera que hagan contacto con otros sin solaparlos.

Capítulo 3: El método de Elementos Discretos.

1.0 - Introducción y aspectos generales de la formulación del método de los elementos discretos.

El Método de los elementos discretos fue introducido por Cundall (1971) para el análisis de problemas de mecánica de rocas y con posterioridad se aplicó a los sólidos por Cundall y Strack (1979) [1]. Esta formulación en los últimos años se ha continuado desarrollando para modelar problemas a nivel de la microestructura de los materiales cubriendo nuevas escalas de la modelación de problemas de la mecánica computacional.

El método de los elementos discretos simula el comportamiento mecánico de un sistema conformado por una colección o sistema de partículas dispuestas arbitrariamente. Este modelo considera las partículas como elementos discretos que en su conjunto conforman el sistema complejo de partículas. Estos elementos distintos, como también se le conoce, se desplazan independientemente uno de otros e interaccionan entre sí en las zonas de contacto.

En este método a nivel de cada partícula se hace uso de la mecánica del cuerpo rígido y los elementos discretos se consideran elementos rígidos en sí. El modelo constitutivo o de comportamiento del material es establecido en las zonas de contactos entre partículas y queda caracterizado por varios elementos mecánicos como: muelles, pistones y elementos de fricción, etc. Los elementos muelles describen la fase de comportamiento elástico del medio en la zona de contacto entre cada partícula. Este comportamiento elástico queda caracterizado por dos muelles: uno en la dirección de contacto normal y otro en la dirección tangencial, los cuales corresponden con la descomposición de fuerzas de contacto que se utilizan en la formulación del método. Los elementos de fricción describen la descohesión y la falla del material en la zona de contacto entre cada partícula. Por su parte los pistones son elementos que toman en cuenta la viscosidad del medio que se simula y son elementos reguladores de la energía. En la formulación establecida pueden emplearse indistintamente varios modelos de contacto viscoso y no viscoso en función del tipo de modelos constitutivo o de contacto que se emplee.

Como el medio es descrito por un sistema de partículas es necesario emplear la ecuación de balance de la cantidad de movimiento. Supóngase para ello de un sistemas discreto formado por n elementos distintos tal que cada partícula i tiene una masa m_i , que se mueve con una aceleración a_i y esta sometida a una fuerza f_i . En este caso la segunda ley de Newton establece que la fuerza que actúa sobre las partículas es igual a la masa de cada elemento discreto por su

aceleración. Utilizando la definición de aceleración como la derivada material de la velocidad y teniendo en cuenta el principio de conservación de la masa (la variación de la masa de la partícula es igual a cero) se tiene:

$$\mathbf{f}_i = m_i \mathbf{a}_i = m_i \frac{d\mathbf{v}_i}{dt} = \frac{d}{dt}(m_i \mathbf{v}_i)$$

Definiendo la cantidad de movimiento de la partícula como el producto de su masa por su velocidad (m_i, \mathbf{v}_i), expresa que la fuerza que actúa sobre el elemento distinto o partícula es igual a la variación de la cantidad de movimiento de la misma. Aplicando este concepto y la segunda Ley de Newton al sistema de n partículas o elementos discretos se tiene:

$$\mathbf{R}(t) = \sum_i \mathbf{f}_i = \sum_i m_i \mathbf{a}_i = \sum_i m_i \frac{d\mathbf{v}_i}{dt} = \frac{d}{dt} \left(\sum_i m_i \mathbf{v}_i \right)$$

suponiendo que se cumple el principio de conservación de la masa $\left(\frac{dm_i}{dt} = 0 \right)$

La ecuación anterior expresa que la resultante de todas las fuerzas que actúan sobre el sistema discreto de partículas es igual a la variación por unidad de tiempo de la cantidad de movimiento del mismo.

La ley de Newton proporciona la relación fundamental entre el movimiento del sistema de partículas y las fuerzas que causan dicho movimiento. El sistema de fuerzas puede estar en equilibrio estático cuando estas no están en movimiento o no actúan fuerzas sobre los diferentes elementos distintos o el medio en cuestión.

Las fuerzas, los desplazamientos, tensiones y deformaciones son determinados a nivel de cada zona de contacto entre los elementos distintos. En correspondencia con el modelo constitutivo empleado, estos elementos mecánicos son descompuestos en sus componentes normal y desviadora.

La formulación establecida parte de una serie de hipótesis que permiten simplificar el problema real desechando los aspectos menos significativos y permitiendo establecer un modelo físico y matemático del problema en estudio. Estas hipótesis son:

1. Las partículas son consideradas como cuerpos rígidos.
2. El contacto ocurre en el punto o área muy pequeña de contacto entre cada partícula.
3. En las uniones entre partículas se considera que existen contactos entre los elementos discretos.

4. Las partículas en 2D son circulares, elípticas, poligonales y colisiones de circunferencias para lograr partículas irregulares. En 3D pueden ser esferas, elipsoides, poliedros y esferas colisionadas para lograr partículas irregulares. Si embargo, la formulación puede considerar o emplear otros tipos de partícula con formas diversas y arbitrarias.
5. La generación del medio empleando elementos discretos o partículas debe ser aleatoria y las dimensiones de los mismos deben ser tratados de forma similar (posición y dimensiones de los elementos distintos aleatorios).
6. Se trabaja en el campo de las pequeñas deformaciones.
7. El comportamiento constitutivo en la zona de contacto emplea una tolerancia (separación / penetración), donde a las partículas o elementos distintos se les permite cierto solape (holgura, gap o penetración) o separación en el punto de contacto. Este aspecto implica desde el punto de vista numérico un contacto aproximado.
8. La magnitud del solape (holgura, penetración o gap) y la separación está relacionada con la fuerza de contacto, la ley fuerza-desplazamiento (modelo constitutivo de contacto), y la magnitud de estos es pequeña con relación al tamaño de los elementos distintos o partículas.

Suponer que las partículas son elementos rígidos es mejor cuando la deformación del sistema físico es considerada a partir del movimiento a lo largo de las interfaces o zonas de contacto entre las partículas.

En la formulación del modelo se han incluido elementos rígidos o paredes, a los cuales, se les puede imponer condiciones de contorno como: velocidades impuestas, desplazamientos, fuerzas o el caso totalmente opuesto como restricciones de movimiento.

En el método de los elementos discretos, la interacción de las partículas es abordada en el proceso de cálculo como un proceso dinámico con desarrollo de estados de equilibrio siempre y cuando exista un balance de las fuerzas interiores. La fuerza de contacto, los desplazamientos, tensiones y deformaciones relativas de una partícula del conjunto son determinadas a través de los movimientos que presentan las partículas individuales. Los movimientos son el resultado de la propagación a través del sistema de partículas de perturbaciones (condiciones de contorno impuestas: fuerzas, desplazamientos, velocidades, etc.) causadas por las paredes o por condiciones impuestas a los mismos elementos discretos. En este proceso dinámico la velocidad de propagación de las perturbaciones impuestas sobre el sistema de partículas depende de las propiedades físicas y mecánicas del sistema discreto en cuestión.

El comportamiento dinámico del sistema de partículas se representa numéricamente por un algoritmo explícito con determinado paso de tiempo considerando que las velocidades y aceleraciones son constantes dentro de cada paso de tiempo. El método de los elementos discretos se basa en la idea de que el paso de tiempo escogido tiene que ser muy pequeño, de forma tal que durante la propagación de una perturbación de una partícula a otra en un paso de tiempo determinado no se puede propagar la misma más allá que a los vecinos inmediatos de cada partícula. Este aspecto propicia que en todo momento, las fuerzas que actúan en cualquier partícula sean exclusivamente determinadas por su interacción con los elementos discretos con los que está en contacto. El paso de tiempo se calcula en función de las propiedades físicas y

mecánicas del medio, lo que permite asegurar que la velocidad a que una perturbación determinada se propaga cumpla con la restricción y suposiciones anteriores.

El proceso de cálculo en el método de los elementos discretos se realiza alternando la aplicación de la segunda ley de Newton y una ley de fuerza-desplazamiento (ecuación constitutiva de contacto) en los contactos existentes entre todos los elementos que conforman el sistema de partículas. La segunda ley de Newton se usa para determinar el movimiento de cada partícula que se origina como resultado de la acción de las fuerzas de contacto y las fuerzas volumétricas, mientras que la ley constitutiva (ley fuerza-desplazamiento) se emplea para actualizar las fuerzas originadas por el movimiento relativo en cada contacto. En los contactos esfera pared solo se requiere aplicar la ley de fuerza-desplazamiento para cada contacto y no es necesario aplicar la ley de Newton porque en el caso de las paredes los movimientos son prefijados como condiciones impuestas.

El modelo consiste en emplear elementos discretos o partículas rígidos para modelar el medio. Esta forma de modelación es apropiada para materiales granulares, geomateriales y otros. La hipótesis de considerar los elementos discretos o partículas rígidos y concentrar la deformación en los puntos de contacto entre ellos es adecuada para describir y caracterizar el comportamiento de los materiales (metales, terreno, rocas, suelo, etc) y el movimiento de las discontinuidades a nivel microscópico y macroscópico. Por consiguiente este modelo también puede aplicarse para modelar el problema a diversas escalas (modelos multiescala) en función de las dimensiones de los diversos elementos que conforman el medio. En la definición de las leyes de contacto entre elementos se incluye las fuerzas de cohesión y fricción, aspectos que permiten modelar la fractura y la descohesión o pérdida de cohesión del material. Esta consideración en el modelo permite a su vez delimitar la formación de micro-fisuras y fisuras que provocan al final el colapso de una estructura. Tener en cuenta en el modelo de contacto la cohesión y fricción del material permite modelar materiales como metales, rocas, suelos, etc., que colapsarán, para ciertos rangos de esfuerzos, debido a la propagación de fisuras en el seno de los mismos.

2.0 - Proceso de cálculo del método de los elementos discretos.

El proceso del cálculo es un algoritmo cíclico o repetitivo que requiere la aplicación de la ley de movimiento a cada partícula, una ley de fuerza-desplazamiento en cada contacto y una actualización constante de posiciones de las paredes. En cada paso de tiempo cambian de forma dinámica la estructura de contacto existente entre los elementos discretos o entre las partículas y las paredes. Este aspecto implica tener un control estricto de los contactos en cada paso de tiempo, además de tener implementado un algoritmo eficiente que actualice constantemente los mismos durante el transcurso de la simulación. De forma muy simplificada el proceso de cálculo se ilustra en la figura siguiente:

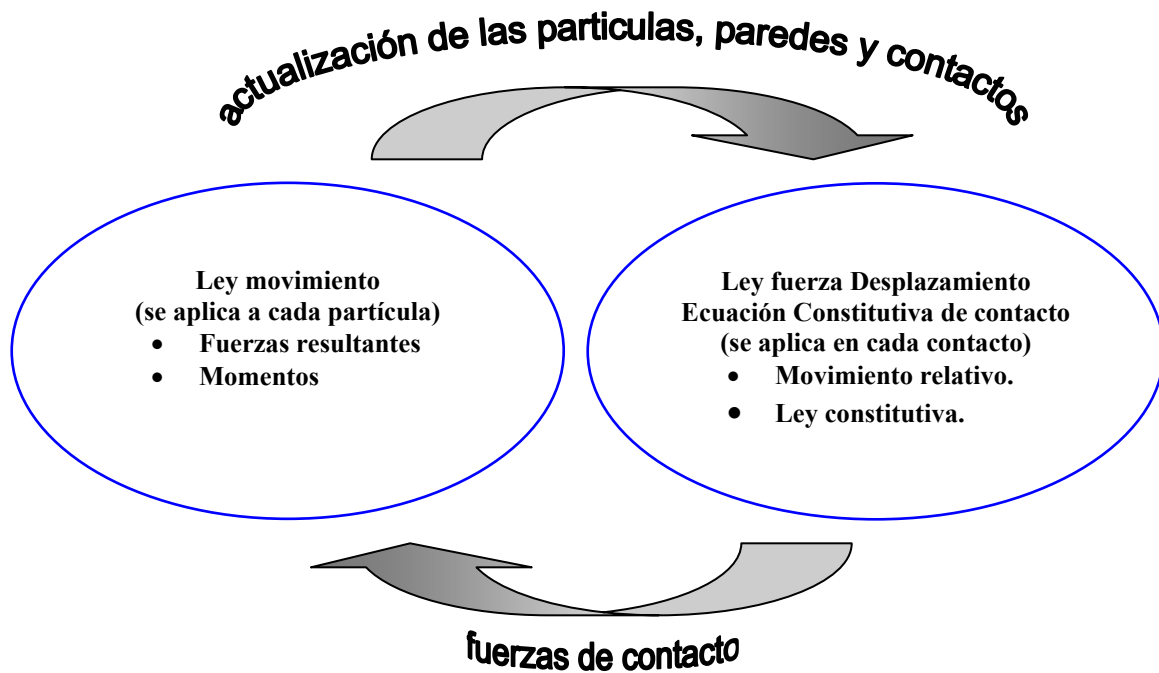


Figura 1: Esquema explícito del método de los elementos discretos.

En el proceso de cálculo en cada instante de tiempo, los contactos se actualizan obteniéndose los diferentes contactos entre partículas y partículas-pared, conociéndose además la posición de cada elemento discreto y de las paredes. Por su parte la ley fuerza-desplazamiento se aplica en cada contacto para actualizar las fuerzas basándose en el movimiento relativo entre las partículas en contacto y el modelo constitutivo empleado. Seguidamente, la ley de movimiento se aplica a cada partícula para actualizar su velocidad y la posición en base a las fuerzas y momentos resultantes que se originan como resultado de la acción de las fuerzas de contacto y las fuerzas volumétricas en cada elemento discreto. Análogamente la posición de la pared se actualiza basándose en las velocidades especificadas como condición impuesta.

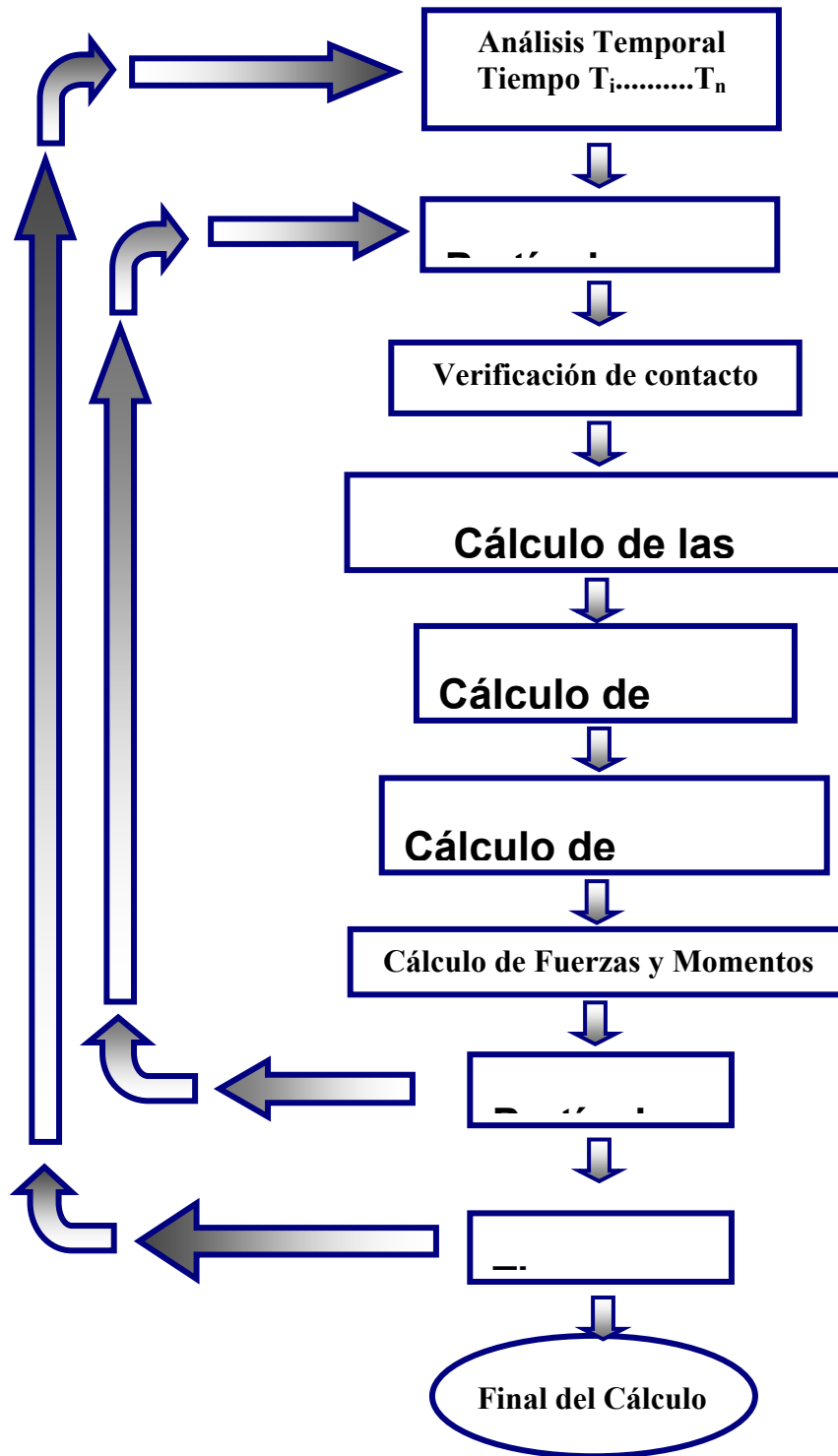


Figura 2: Esquema de la secuencia de cálculo en el método de los elementos discretos.

3.0 - Formulación del método de los elementos discretos.

3.1 – Ecuaciones de Movimiento.

El estudio de modelos de partículas a través del Método de los Elementos Discretos o de Partículas con el empleo de nuevos modelos constitutivos micro-estructurales es un tema de indudable interés científico en el campo de los métodos numéricos en la ingeniería.

La finalidad de acercarse cada vez más al micro-mundo y a la esencia micro-estructural de los fenómenos físicos, permite realizar estudios de seguridad estructural con un alto grado de precisión y fiabilidad, lo que a su vez posibilita prever fallos estructurales severos, ya que los modelos discretos (modelos de partículas) permiten detectar fallos a nivel micro. El hecho de poder estudiar los estados tensionales y deformacionales a un nivel microscópico posibilita detectar la aparición de microfisuras y discontinuidades que son el comienzo de las cadenas de fallos estructurales y la formación de micro-zonas de plastificación que culminan desafortunadamente, a nivel macroscópico, en el fallo estructural. Estas delimitaciones no apreciables por los métodos macroscópicos convencionales, posibilita estudiar la evolución de las micro-grietas y permite a su vez prevenir catástrofes y desastres que pueden afectar a las estructuras.

El movimiento de los diferentes elementos discretos que conforman el modelo de partículas se rige por las leyes de la dinámica de la mecánica del sólido rígido. La traslación y la rotación de las partículas rígidas se describen por medio de las ecuaciones de la dinámica de Newton-Euler (figura 3). El movimiento del elemento i -ésimo del conjunto de elementos discretos se describe por las siguientes ecuaciones:

$$\mathbf{m}_i \ddot{\mathbf{u}}_i = \mathbf{F}_i \quad (1)$$

$$\mathbf{I}_i \dot{\boldsymbol{\omega}}_i = \mathbf{T}_i \quad (2)$$

donde:

- u** - es el vector de desplazamiento del centroide del elemento en un sistema de coordenadas fijo **X**.
- ω** - velocidad angular del elemento respecto al sistema de referencia móvil **X**, con el elemento y el origen en el centro del mismo.
- m** - masa del elemento o partícula i -ésimo.
- I** - momento de inercia.
- F** - fuerza resultante
- T** - momento resultante.

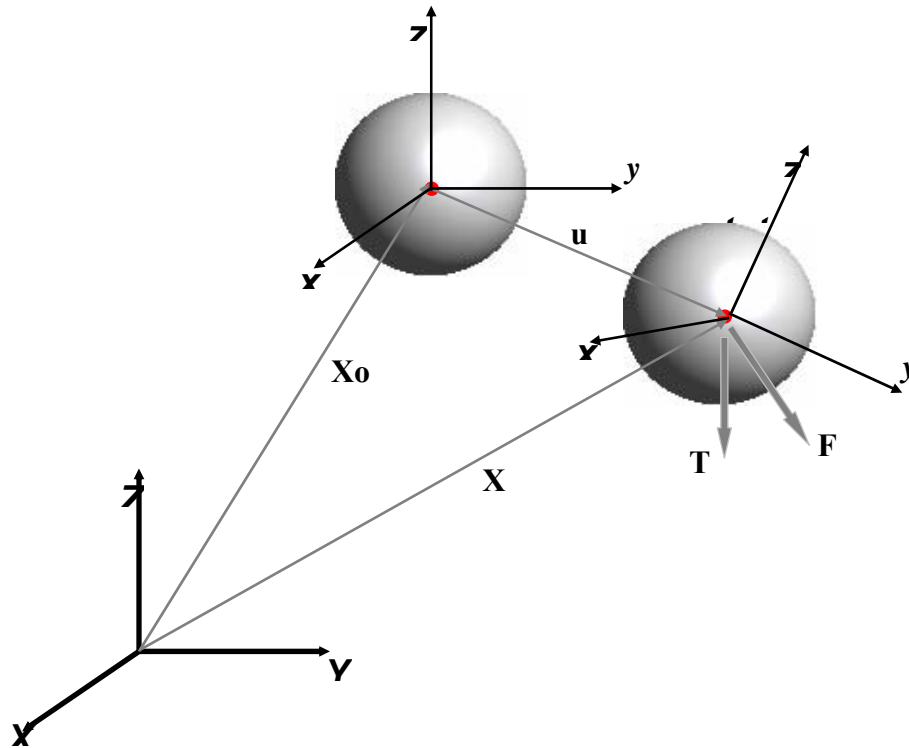


Figura 3: Movimiento de sólido rígido en un sistema de referencia espacial

3.2 - Amortiguamiento global.

Los códigos dinámicos explícitos satisfacen mejor la simulación de procesos dinámicos, aunque se pueden aplicar con facilidad en la simulación de problemas cuasi-estáticos. Una de las metodologías elaboradas para analizar los problemas cuasi-estáticos con el modelo dinámico explícito es la relajación dinámica [2]. Un factor importante para el éxito del análisis de los problemas cuasi-estáticos con un código dinámico consiste en la aplicación de un amortiguamiento adecuado. En el modelo de los elementos discretos, las oscilaciones con las frecuencias más altas se amortiguan con el amortiguamiento impuesto al contacto. Igualmente en los problemas cuasi-estáticos, es necesario amortiguar las oscilaciones de los modos de vibración más bajos a través de un amortiguamiento exterior o global. Esto puede lograrse aplicando el amortiguamiento viscoso global cerca del valor crítico para las frecuencias más bajas. La velocidad de las cargas también debe corresponder a las propiedades dinámicas del sistema, y debe introducirse durante un tiempo suficientemente largo comparado con el período de vibraciones en el modo natural más bajo.

El estado de equilibrio del sistema de partículas puede lograrse por la aplicación de un amortiguamiento adecuado. A veces es necesaria la aplicación de un amortiguamiento para disipar la energía cinética de las partículas que no están en contacto y esta es la justificación para el empleo de un amortiguamiento global del sistema de partículas. En el caso del amortiguamiento global, al igual que en el amortiguamiento de contacto, existen dos tipos:

amortiguamiento viscoso y no viscoso. En los dos casos los términos de amortiguamiento $F_i^{Amortig}$ y $T_i^{Amortig}$ se agregan a las ecuaciones de movimiento (1) y (2):

$$m_i \ddot{u}_i = F_i + F_i^{Amortig} \quad (3)$$

$$I_i \dot{\omega}_i = T_i + T_i^{Amortig} \quad (4)$$

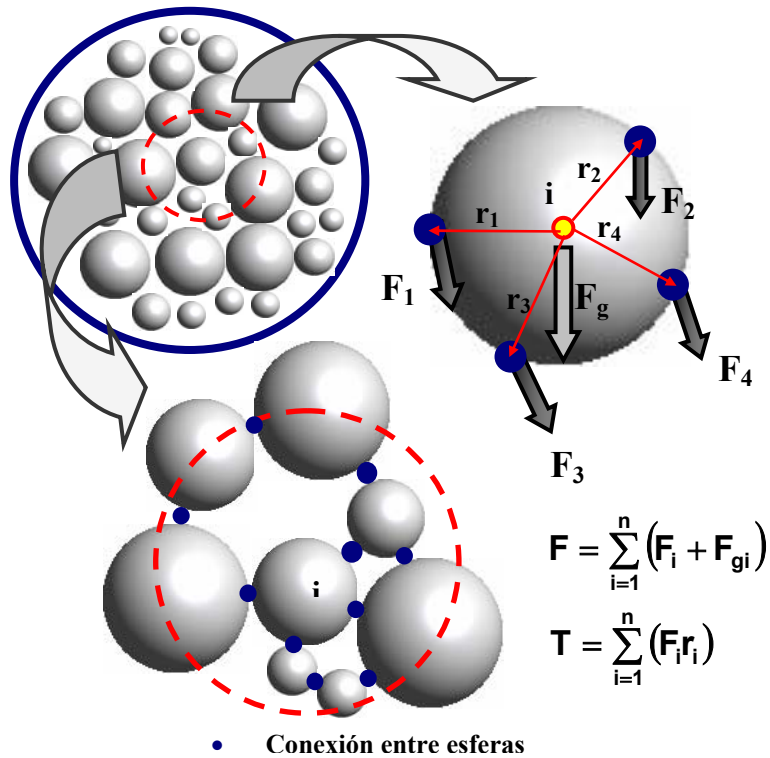


Figura 4 : Fuerzas de contacto entre partículas o elementos discretos.

El cálculo del amortiguamiento global es diferente para el caso viscoso y el no viscoso. En el viscoso el amortiguamiento queda definido por:

$$F_i^{amortig} = -\alpha^{VT} m_i \dot{u}_i$$

$$T_i^{amortig} = -\alpha^{Vr} I_i \dot{\omega}_i$$

y para el caso no viscoso :

$$\mathbf{F}_i^{\text{amortig}} = -\alpha^{nVT} \|\mathbf{F}_i\| \frac{\dot{\mathbf{u}}_i}{\|\dot{\mathbf{u}}_i\|}$$

$$\mathbf{T}_i^{\text{amortig}} = -\alpha^{nVr} \|\mathbf{T}_i\| \frac{\dot{\boldsymbol{\omega}}_i}{\|\dot{\boldsymbol{\omega}}_i\|}$$

donde: α^{vT} , α^{vr} , α^{nvT} , $\alpha^{nv r}$ son las constantes de amortiguamiento. En estas ecuaciones queda expresado el amortiguamiento del sistema global para los casos viscoso y no viscoso. En el caso de amortiguamiento viscoso la fuerza de amortiguamiento es proporcional a la velocidad, mientras que en el no viscoso es proporcional a la fuerza y al momento resultante.

3.3 - Determinación de la densidad y masa micro-estructural.

La masa de cada elemento discreto se puede estimar conociendo de antemano la densidad de material y las dimensiones de cada una de las partículas que conforman el medio. En el caso de partículas esféricas por ejemplo, el cálculo de la masa (figura 5) puede realizar por:

$$\mathbf{m}_i = \gamma \mathbf{V}_i$$

donde:

γ - densidad del material.

\mathbf{V} - Volumen.

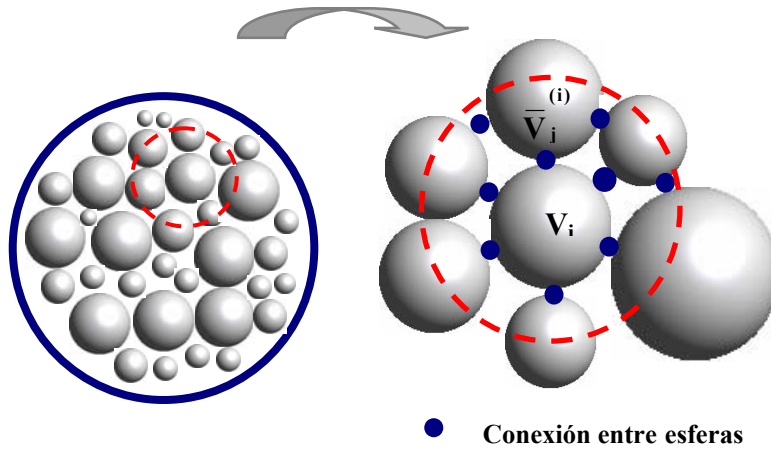


Figura 5: Definición del volumen de control para evaluar la densidad microestructural

En los modelos de partículas, como su nombre lo indica, se discretiza el medio por elementos discretos, lo que propicia la formación de zonas de vacío o huecos que a su vez delimitan la porosidad del material. Esto hace necesario tomar en cuenta este aspecto para el cálculo de la densidad micro-estructural. Para ello en cada partícula se define un volumen de control (V_c), lo que posibilita definir la densidad micro para cada elemento discreto. Este aspecto en el caso de elementos esféricos se formula por:

$$\bar{\gamma}^{(i)} = \gamma \frac{\mathbf{V}_c - \mathbf{V}_o^{(i)}}{\mathbf{V}_c}, \quad \mathbf{V}_o^{(i)} = \mathbf{V}_c - \mathbf{V}_i - \sum \mathbf{V}_j^{(i)}$$

$$\mathbf{V}_i = \frac{4}{3} \pi r_i^3 \quad (3D - \text{esferas})$$

Para el caso 2D y con partículas cilíndricas, se trabaja con el concepto de área de control (A_c) obteniéndose:

$$\bar{\gamma}^{(i)} = \gamma \frac{\mathbf{A}_c - \mathbf{A}_o^{(i)}}{\mathbf{A}_c}, \quad \mathbf{A}_o^{(i)} = \mathbf{A}_c - \mathbf{A}_i - \sum \mathbf{A}_j^{(i)}$$

$$\mathbf{A}_i = \pi r^2 \quad (2D - \text{cilindros})$$

3. 4 - Integración del problema con un esquema de diferencia finita central.

El movimiento rotatorio se describe con respecto al corrotacional de un sistema de coordenadas locales x incluido en cada elemento. Los vectores \mathbf{F} y \mathbf{T} son las sumas de todas las fuerzas y momentos aplicadas en el elemento i -ésimo debido a la carga externa, y a las interacciones de contacto entre las partículas, etc., así como a las fuerzas resultante del amortiguamiento en el sistema.

Las ecuaciones de movimiento (1), (2) o (3) (4) se integran en el tiempo usando un esquema de diferencias finitas centrado. La integración del movimiento traslacional de los elementos discretos para el paso de tiempo i -ésimo se realiza como sigue:

$$\ddot{\mathbf{u}}_i^n = \frac{\mathbf{F}_i^n}{m_i} \quad (5)$$

Tomando en cuenta las fuerzas de amortiguamiento:

$$\ddot{\mathbf{u}}_i^n = \frac{\mathbf{F}_i^{\text{amortig}} + \mathbf{F}_i^n}{m_i} \quad (6)$$

$$\dot{\mathbf{u}}_i^{n+\frac{1}{2}} = \dot{\mathbf{u}}_i^{n-\frac{1}{2}} + \ddot{\mathbf{u}}_i^n \Delta t \quad (7)$$

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \dot{\mathbf{u}}_i^{n+\frac{1}{2}} \Delta t \quad (8)$$

En el caso del movimiento rotacional también se emplea un esquema en diferencias finitas centrado, obteniéndose para la velocidad y el desplazamiento angular:

$$\dot{\omega}_i^n = \frac{\mathbf{T}_i^n}{I_i} \quad (9)$$

y tomando en cuenta el amortiguamiento global:

$$\dot{\omega}_i^n = \frac{\mathbf{T}_i^{\text{amortig}} + \mathbf{T}_i^n}{I_i} \quad (10)$$

$$\omega_i^{n+\frac{1}{2}} = \omega_i^{n-\frac{1}{2}} + \dot{\omega}_i^n \Delta t \quad (11)$$

En problemas en 2D el ángulo de la rotación (θ) se puede obtener de modo similar al vector de los desplazamientos \mathbf{u}_i (8):

$$\theta_i^{n+1} = \theta_i^n + \omega_i^{n+\frac{1}{2}} \Delta t \quad (12)$$

En el caso del movimiento tridimensional (3D), la matriz de rotación define el cambio de coordenadas del sistema de referencia móvil en cada paso de tiempo \mathbf{x}_i respecto al sistema de referencia fijo \mathbf{X}_i :

$$\mathbf{X}_i = \mathbf{\Lambda}_i \mathbf{x}_i \quad (13)$$

La matriz de rotación se formula para cada paso de tiempo [3], [4] por:

$$\Delta \theta_i = \omega_i^{n+\frac{1}{2}} \Delta t \quad (14)$$

$$\Delta \mathbf{\Lambda}_i = \cos \|\Delta \theta_i\| \mathbf{1} + \frac{\sin \|\Delta \theta_i\|}{\|\Delta \theta_i\|} \Delta \tilde{\theta}_i + \frac{1 - \cos \|\Delta \theta_i\|}{\|\Delta \theta_i\|^2} \Delta \theta_i \Delta \theta_i^T \quad (15)$$

$$\mathbf{\Lambda}_i^{n+1} = \Delta \mathbf{\Lambda}_i \mathbf{\Lambda}_i^n \quad (16)$$

donde:

$\Delta \theta_i = \{\Delta \theta_x, \Delta \theta_y, \Delta \theta_z\}^T$ denota el vector de rotación incremental,

$\Delta\Lambda_i$ es la matriz rotacional incremental,
 $\Delta\tilde{\theta}_i$ es una matriz antisimétrica, definida como:

$$\Delta\tilde{\theta} = \begin{bmatrix} \mathbf{0} & -\Delta\theta_z & \Delta\theta_y \\ \Delta\theta_z & \mathbf{0} & -\Delta\theta_x \\ -\Delta\theta_y & \Delta\theta_x & \mathbf{0} \end{bmatrix} \quad (17)$$

3.5 - Estabilidad numérica del esquema de integración explícito.

La integración explícita en el tiempo presenta una alta eficiencia computacional pero no es incondicionalmente estable. La desventaja del esquema de la integración explícito es conocida, su estabilidad numérica condicional impone una limitación en el paso o discretización del tiempo. Para cumplir con la estabilidad del método es necesario que el paso de tiempo sea menor o igual al tiempo crítico:

$$\Delta t \leq \Delta t_{\text{critico}} \quad (18)$$

El tiempo crítico se obtiene a partir de la frecuencia natural más alta del sistema ω_{\max} por:

$$\Delta t_{\text{critico}} = \frac{2}{\omega_{\max}} \quad (19)$$

Si existe amortiguamiento, el incremento de tiempo crítico se calcula por:

$$\Delta t_{\text{critico}} = \frac{2}{\omega_{\max}} \left(\sqrt{1 + \xi^2} - \xi \right) \quad (20)$$

donde:

ξ - es la fracción del amortiguamiento crítico que corresponde a la frecuencia más alta ω_{\max} .

La determinación exacta de la frecuencia más alta ω_{\max} requiere la solución de un problema de autovalores definidos para el sistema de n partículas rígidas conectadas entre sí. En un procedimiento de solución aproximado pueden definirse los problemas de autovalores separadamente usando las ecuaciones linealizadas del movimiento:

$$\mathbf{m}_i \ddot{\mathbf{r}}_i + \mathbf{k}_i \mathbf{r}_i = \mathbf{0} \quad (21)$$

donde

$$\mathbf{m}_i = \{m_i, m_i, m_i, I_i, I_i, I_i\}^T, \mathbf{r}_i = \{(u_x)_i, (u_y)_i, (u_z)_i, (\theta_x)_i, (\theta_y)_i, (\theta_z)_i\}^T \quad (22)$$

\mathbf{k}_i - matriz rigidez que toma en cuenta la contribución de todas las partículas i .

La ecuación (22) define los vectores \mathbf{m}_i y \mathbf{r}_i para una partícula esférica en el espacio tridimensional. Para una partícula cilíndrica en un modelo bidimensional se definen como sigue:

$$\mathbf{m}_i = \{m_i, m_i, I_i\}^T, \quad \mathbf{r}_i = \{(u_x)_i, (u_y)_i, (\theta_z)_i\}^T \quad (23)$$

La ecuación (21) conduce a resolver el problema de autovalores siguiente:

$$\mathbf{k}_i \mathbf{r}_i = \mathbf{m}_i \lambda_i \mathbf{r}_i \quad (24)$$

donde: los autovalores λ_j ($j \in \{1, \dots, 6\}$ en el caso de 3D, y para el caso de 2D $j \in \{1, \dots, 3\}$) son los cuadrados de las frecuencias de vibraciones libres:

$$\lambda_j = \omega_j^2 \quad (25)$$

En los problemas 3D, tres de las seis frecuencias son de translación, y las otras tres de rotación, mientras que en 2D dos frecuencias son de translación y una de rotación.

Una forma de simplificar el problema de autovalores para el cálculo de la frecuencia máxima ω_{\max} puede ser estimando la misma como la máxima frecuencia natural de un sistema masa-resorte definido para todas las partículas con una translación y una rotación por grado de libertad. Las traslaciones y las rotaciones de la vibración libre son gobernadas por las ecuaciones siguientes:

$$\mathbf{m}_i \ddot{\mathbf{u}}_{ni} + \mathbf{k}_n \mathbf{u}_{ni} = \mathbf{0} \quad (26)$$

$$I_i \ddot{\theta}_{ni} + \mathbf{k}_\theta \mathbf{u}_{ni} = \mathbf{0} \quad (27)$$

donde se tiene que el movimiento de translación es debido a la interacción de contacto en la dirección normal (\mathbf{k}_n es la rigidez elástica del contacto en la dirección normal), y el movimiento rotacional a la interacción en el sentido tangencial (\mathbf{k}_T rigidez elástica del contacto en la dirección tangencial al mismo). Conocida la rigidez tangencial, la rigidez rotatoria puede obtenerse como:

$$\mathbf{k}_\theta = \mathbf{k}_T \mathbf{r}^2 \quad (28)$$

donde \mathbf{r} es la longitud del vector que conecta el centro de masa de los puntos en contacto

La frecuencia natural de las vibraciones de translación se obtiene por la ecuación siguiente:

$$\omega_n = \sqrt{\frac{k_n}{m_i}} \quad (29)$$

mientras la frecuencia rotacional puede obtenerse por la fórmula

$$\omega_\theta = \sqrt{\frac{k_\theta}{I_i}} \quad (30)$$

Como la inercia rotacional en el caso de partículas cilíndricas y esféricas es:

$$I = \frac{2}{5} m r^2 \quad (3D, \text{esferas}) \quad (31)$$

$$I = \frac{m r^2}{2} \quad (2D, \text{discos}) \quad (32)$$

y k_θ esta dado por la ecuación (28), la frecuencia rotacional puede calcularse para el caso de cilindros y esferas como:

$$\omega_\theta = \sqrt{\frac{5k_T}{2m_i}} \quad (3D, \text{esferas}) \quad (33)$$

$$\omega_\theta = \sqrt{\frac{2k_T}{m_i}} \quad (2D, \text{discos}) \quad (34)$$

Si $k_T = k_n$ la frecuencia rotacional ω_θ es considerablemente mayor que la frecuencia de translación ω_n obtenida por la ecuación (30), como resultado el incremento de tiempo crítico tiene que ser menor que el obtenido por la ecuación (19). Con tal de evitar que el paso de tiempo crítico venga fijado por las frecuencias de rotación, la inercia rotacional se escala de forma adecuada.

3.6 - Especificidades de la lógica de cluster de esferas o conglomerados.

En el caso de conglomerados o cluster de esferas para lograr partículas irregulares existen un grupo de especificidades. En este caso para dejar clara la notación empleada en lo adelante se parte de que todas las cantidades vectoriales y tensoriales son expresadas usando notación indicial con respecto a un sistema de coordenadas rectangulares cartesianas. Así, el vector de posición es denotado por \mathbf{x}_i , y el tensor de tensiones es denotado por σ_{ij} , donde se asume que todos los índices corren sobre el conjunto $\{1, 2, 3\}$ para el caso tridimensional y el conjunto $\{1, 2\}$ para el caso bidimensional. El convenio de sumatoria de Einstein es empleado; así, la repetición de un índice en un término denota una sumatoria con respecto a ese índice sobre su rango. Por ejemplo, el vector de tracción \mathbf{t}_i , actuando en la dirección \mathbf{n}_i , está dado por $\mathbf{t}_i = \sigma_{ij} \mathbf{n}_j = \sigma_{i1} \mathbf{n}_1 + \sigma_{i2} \mathbf{n}_2 + \sigma_{i3} \mathbf{n}_3$. Barras verticales denotan la magnitud de un vector o el valor absoluto de un escalar.

Lógica del conglomerado

La lógica del conglomerado provee medios para crear y modificar grupos de partículas esclavas o *conglomerados*. Un conglomerado se comporta como un cuerpo rígido (i.e., las partículas componentes del conglomerado permanecen a una distancia fija de cada una). Los contactos internos al conglomerado son obviados durante el ciclo de cálculo, resultando en un ahorro de tiempo de cómputo comparado a un cálculo similar en el cual todos los contactos están activos. Sin embargo, los contactos con partículas externas al conglomerado no son afectados (i.e., tales contactos se desarrollarán cuando las partículas componentes de la frontera de un conglomerado vienen a hacer contacto con otras partículas). Las partículas dentro de un conglomerado pueden solaparse en cualquier medida; las fuerzas de contacto que existen cuando el conglomerado es creado o cuando una partícula es agregada al conglomerado serán preservadas invariables durante el ciclo. Así, un conglomerado actúa como un cuerpo rígido (con una frontera deformable) que no se romperá en pedazos, independientemente de las fuerzas que actúen sobre él. En este sentido, un conglomerado difiere de un grupo de partículas que están enlazadas mutuamente.

3.6.1 - Propiedades de masa de un conglomerado.

Las propiedades básicas de masa de un conglomerado son su masa total \mathbf{m} , ubicación de su centro de masa $\mathbf{u}_i^{[G]}$, y momentos y productos de inercia I_{ii} e I_{ij} . Para un conglomerado general compuesto por N_p bolas, cada una con masa $\mathbf{m}^{[p]}$, radio $\mathbf{R}^{[p]}$ y localización del centroide $\mathbf{u}_i^{[p]}$, las propiedades de masa están definidas por las siguientes ecuaciones:

$$\mathbf{m} = \sum_{p=1}^{N_p} \mathbf{m}^{[p]} \quad (35)$$

$$\mathbf{u}_i^{[G]} = \frac{1}{\mathbf{m}} \sum_{p=1}^{N_p} \mathbf{m}^{[p]} \mathbf{u}_i^{[p]} \quad (36)$$

$$I_{ii} = \sum_{p=1}^{N_p} \left\{ \mathbf{m}^{[p]} (\mathbf{u}_j^{[p]} - \mathbf{u}_j^{[G]})(\mathbf{u}_j^{[p]} - \mathbf{u}_j^{[G]}) + \frac{2}{5} \mathbf{m}^{[p]} \mathbf{R}^{[p]} \mathbf{R}^{[p]} \right\}; \quad (j = i)$$
$$I_{ij} = \sum_{p=1}^{N_p} \left\{ \mathbf{m}^{[p]} (\mathbf{u}_i^{[p]} - \mathbf{u}_i^{[G]})(\mathbf{u}_j^{[p]} - \mathbf{u}_j^{[G]}) \right\}; \quad (j \neq i) \quad (37)$$

Los momentos y productos de inercia están definidos con respecto a un marco de referencia que está fijado al conglomerado en su centro de masa y alineado con el sistema global de ejes. En general, este será un sistema de ejes no principales (i.e., $I_{ij} \neq 0$).

3.6.2 Ecuaciones completas de movimiento para un conglomerado.

El movimiento de un conglomerado está determinado por la fuerza resultante y momentos vectoriales actuando sobre él. Como el conglomerado es tratado como un cuerpo rígido, su movimiento puede ser descrito en términos del movimiento traslacional de un punto en el conglomerado y el movimiento rotacional de todo el conglomerado. El movimiento traslacional del centro de masas está descrito en términos de su posición \mathbf{u}_i , velocidad $\dot{\mathbf{u}}_i$, y aceleración $\ddot{\mathbf{u}}_i$; el

movimiento rotacional del conglomerado es descrito en términos de su velocidad angular ω_i y aceleración angular $\dot{\omega}_i$.

Las ecuaciones de movimiento pueden ser expresadas como dos ecuaciones vectoriales, una de las cuales relaciona la fuerza resultante al movimiento traslacional mientras la otra relaciona el momento resultante al movimiento rotacional. La ecuación para el movimiento traslacional puede ser escrita en la forma vectorial

$$\mathbf{F}_i = m(\ddot{\mathbf{u}}_i - \mathbf{g}_i) \quad (38)$$

en la cual \mathbf{F}_i es la fuerza resultante, la suma de todas las fuerzas externamente aplicadas actuando en el conglomerado, m es la masa total del conglomerado, y \mathbf{g}_i es la fuerza de aceleración del cuerpo que surge por la carga de gravedad. La fuerza resultante es computada por :

$$\mathbf{F}_i = \sum_{p=1}^{N_p} (\tilde{\mathbf{F}}_i^{[p]} + \sum_{c=1}^{N_c} \mathbf{F}_i^{[p,c]}) \quad (39)$$

en la cual $\tilde{\mathbf{F}}_i^{[p]}$ es la fuerza aplicada externamente actuando en la partícula (p) , y $\mathbf{F}_i^{[p,c]}$ es la fuerza actuando en la partícula (p) en el contacto (c) .

La ecuación para el movimiento rotacional puede ser escrita en la forma vectorial [Ginsberg y Genin, 1984]

$$\mathbf{M}_i = \dot{\mathbf{H}}_i \quad (40)$$

En la cual \mathbf{M}_i es el momento resultante respecto al centro de masa, y $\dot{\mathbf{H}}_i$ es la razón de cambio temporal del momento angular del conglomerado.

El momento resultante en la ecuación (40) es computado por

$$\mathbf{M}_i = \sum_{p=1}^{N_p} (\tilde{\mathbf{M}}_i^{[p]} + \varepsilon_{ijk} (\mathbf{x}_j^{[p]} - \mathbf{x}_j^{[G]}) \mathbf{F}_k^{[p]} + \sum_{c=1}^{N_c} \varepsilon_{ijk} (\mathbf{x}_j^{[c]} - \mathbf{x}_j^{[p]}) \mathbf{F}_k^{[p,c]}) \quad (41)$$

en la cual $\tilde{\mathbf{M}}_i^{[p]}$ es el momento aplicado externamente en la partícula (p) , $\mathbf{F}_k^{[p]}$ es la fuerza resultante actuando en la partícula (p) en su centroide, y $\mathbf{F}_k^{[p,c]}$ es la fuerza actuando en la partícula (p) en el contacto (c) .

La ecuación (40) está referida a un sistema local de coordenadas que está fijo al conglomerado en su centro de masa. Para tal sistema, la razón temporal de cambio del momento angular puede ser escrita como

$$\dot{\mathbf{H}}_i = \alpha_i \mathbf{l}_{ii} - \alpha_j \mathbf{l}_{ij} + \varepsilon_{ijk} \omega_j (\omega_k \mathbf{l}_{kk} - \omega_l \mathbf{l}_{kl}); \quad (j \neq i, l \neq k) \quad (42)$$

donde $\alpha_i = \dot{\omega}_i$.

Las ecuaciones de movimiento, dadas por (38) y (40), son integradas usando un procedimiento de diferencias finitas centrado teniendo un paso de tiempo Δt . Las cantidades $\dot{\mathbf{u}}_i$ y ω_i son computadas en los medio intervalos de $\mathbf{t} \pm \frac{n\Delta t}{2}$, mientras las cantidades \mathbf{u}_i , $\ddot{\mathbf{u}}_i$, $\dot{\omega}_i$, \mathbf{F}_i y \mathbf{M}_i son computadas en los intervalos primarios de $\mathbf{t} \pm n\Delta t$.

Las siguientes expresiones de diferencias finitas describen las aceleraciones traslacional y rotacional en el tiempo \mathbf{t} en términos de los valores de velocidad en los mediotintervalos. Las aceleraciones son calculadas como

$$\begin{aligned} \ddot{\mathbf{u}}_i^{(t)} &= \frac{1}{\Delta t} (\dot{\mathbf{u}}_i^{(t+\Delta t/2)} - \dot{\mathbf{u}}_i^{(t-\Delta t/2)}) \\ \dot{\omega}_i^{(t)} &= \frac{1}{\Delta t} \left(\omega_i^{(t+\Delta t/2)} - \omega_i^{(t-\Delta t/2)} \right) \end{aligned} \quad (43)$$

Insertando la primera de estas expresiones en la ecuación (38) y resolviendo para la velocidad en el tiempo $\mathbf{t} + \frac{n\Delta t}{2}$ resulta en

$$\dot{\mathbf{u}}_i^{(t+\Delta t/2)} = \dot{\mathbf{u}}_i^{(t-\Delta t/2)} + \left(\frac{\mathbf{F}_i^{(t)}}{\mathbf{m}} + \mathbf{g}_i \right) \Delta t \quad (44)$$

Obteniendo la velocidad angular en el tiempo $\mathbf{t} + \frac{n\Delta t}{2}$ requiere resolver un conjunto de ecuaciones no lineales usando un procedimiento iterativo durante cada paso de tiempo. Sustituyendo la ecuación (41) en la ecuación (40) permite escribir

$$\mathbf{M}_i - \varepsilon_{ijk} \omega_j (\omega_k \mathbf{l}_{kk} - \omega_l \mathbf{l}_{kl}) = \alpha_i \mathbf{l}_{ii} - \alpha_j \mathbf{l}_{ij}; \quad (j \neq i, l \neq k) \quad (45)$$

la cual, puede ser expresada en forma matricial como

$$\{\mathbf{M}\} - \{\mathbf{W}\} = [\mathbf{I}]\{\alpha\} \quad (46)$$

con

$$\begin{aligned}\{\mathbf{M}\} &= \begin{Bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \\ \mathbf{M}_3 \end{Bmatrix} \\ \{\mathbf{W}\} &= \begin{Bmatrix} \omega_2 \omega_3 (I_{33} - I_{22}) + \omega_3 \omega_3 I_{23} - \omega_2 \omega_2 I_{32} - \omega_1 \omega_2 I_{31} - \omega_1 \omega_3 I_{21} \\ \omega_3 \omega_1 (I_{11} - I_{33}) + \omega_1 \omega_1 I_{31} - \omega_3 \omega_3 I_{13} - \omega_2 \omega_3 I_{12} - \omega_2 \omega_1 I_{32} \\ \omega_1 \omega_2 (I_{22} - I_{11}) + \omega_3 \omega_3 I_{12} - \omega_1 \omega_1 I_{21} - \omega_3 \omega_1 I_{23} - \omega_3 \omega_2 I_{13} \end{Bmatrix} \\ [\mathbf{I}] &= \begin{bmatrix} I_{11} & -I_{12} & -I_{13} \\ -I_{21} & I_{22} & -I_{23} \\ -I_{31} & -I_{32} & I_{33} \end{bmatrix} \\ \{\boldsymbol{\alpha}\} &= \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{Bmatrix} = \begin{Bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{Bmatrix}\end{aligned}$$

La ecuación (46) provee tres ecuaciones para las seis incógnitas, ω_i y α_i . Estas seis incógnitas se encuentran usando el siguiente procedimiento iterativo:

1. Hacer $\mathbf{n} = \mathbf{0}$.
2. Hacer $\omega_i^{[0]}$ igual a la velocidad angular inicial (i.e., antes del cómputo de movimiento).
3. Resolver la ecuación (46) para $\alpha_i^{[n]}$.
4. Determinar una nueva velocidad angular (provisional) asumiendo que no hay amortiguamiento

$$\omega_i^{[nueva]} = \omega_i^{[0]} + \alpha_i^{[n]} \Delta t \quad (47)$$

5. Actualizar el estimado de ω_i como

$$\omega_i^{[n+1]} = \frac{\omega_i^{[0]} + \omega_i^{[nueva]}}{2} \quad (48)$$

6. Hacer $\mathbf{n} := \mathbf{n} + \mathbf{1}$ e ir al paso 3.

La experimentación numérica ha probado que la iteración tiende a converger después de cuatro iteraciones; así, después que cuatro iteraciones hayan sido realizadas, se hace se detiene si uno o más grados rotacionales de libertad están fijos; para tales casos, el método de solución aproximada descrito en la sección 3.6.3 es empleado.

La misma formulación local de amortiguamiento no viscoso usada para disipar energía de cada partícula individual es también usada para disipar energía de cada conglomerado. La formulación

está descrita en investigaciones desarrolladas por Itasca [5]. El coeficiente de amortiguamiento para un conglomerado es la media aritmética de los coeficientes de sus partículas componentes, y la velocidad angular usada en la formulación de amortiguamiento es tomada como $\alpha_i^{[3]}\Delta t$.

Después de determinar las velocidades traslacional y rotacional del conglomerado, la posición del centro de masas del conglomerado es actualizada mediante

$$\mathbf{u}_i^{(t+\Delta t)} = \mathbf{u}_i^{(t)} + \dot{\mathbf{u}}_i^{(t+\Delta t/2)}\Delta t \quad (49)$$

y la velocidad de cada bola en el conglomerado es puesta como

$$\dot{\mathbf{u}}_i^{[p]} = \dot{\mathbf{u}}_i^{[G]} + \varepsilon_{ijk}\omega_j(\mathbf{u}_k^{[p]} - \mathbf{u}_k^{[G]}) \quad (50)$$

(no centrada en el tiempo)

Cada bola en el conglomerado es entonces actualizada basándose en la velocidad de las bolas de acuerdo a la ecuación (49).

La ecuación (50) usa vectores de posición de las partículas y el conglomerado que existen un medio paso de tiempo detrás del vector de velocidad calculado. Esto causa una gradual deriva en las localizaciones de las partículas dentro de un conglomerado: el error se relaciona con la magnitud de los componentes de rotación. Para conglomerados que giran rápidamente, es posible obtener un resultado más exacto mediante el uso de vectores de posición centrados en el tiempo en el cálculo de las velocidades de las partículas. Se reemplaza la ecuación (50) con la siguiente expresión:

$$\dot{\mathbf{u}}_i^{[p]} = \dot{\mathbf{u}}_i^{[G]} + \varepsilon_{ijk}\omega_j(\bar{\mathbf{u}}_k^{[p]} - \bar{\mathbf{u}}_k^{[G]}) \quad (51)$$

(centrada en el tiempo)

donde

$$\bar{\mathbf{u}}_i^{[p]} = \mathbf{u}_i^{[p]_0} + \frac{\dot{\mathbf{u}}_i^{[p]}\Delta t}{2} \quad (52)$$

$$\bar{\mathbf{u}}_i^{[G]} = \frac{\mathbf{u}_i^{(t+\Delta t)} + \mathbf{u}_i^{(t)}}{2} \quad (53)$$

y $\mathbf{u}_i^{[p]_0}$ denota el vector de posición de la partícula $[p]$ antes de la actualización.

Expandiendo la ecuación (51) y sustituyendo de la ecuación (52), se obtiene

$$\begin{aligned}
\dot{\mathbf{u}}_1^{[p]} &= \dot{\mathbf{u}}_1^{[G]} + \omega_2 \left(\mathbf{u}_3^{[p]_0} + \dot{\mathbf{u}}_3^{[p]} \Delta t / 2 - \bar{\mathbf{u}}_3^{[G]} \right) - \omega_3 \left(\mathbf{u}_2^{[p]_0} + \dot{\mathbf{u}}_2^{[p]} \Delta t / 2 - \bar{\mathbf{u}}_2^{[G]} \right) \\
\dot{\mathbf{u}}_2^{[p]} &= \dot{\mathbf{u}}_2^{[G]} + \omega_3 \left(\mathbf{u}_1^{[p]_0} + \dot{\mathbf{u}}_1^{[p]} \Delta t / 2 - \bar{\mathbf{u}}_1^{[G]} \right) - \omega_1 \left(\mathbf{u}_3^{[p]_0} + \dot{\mathbf{u}}_3^{[p]} \Delta t / 2 - \bar{\mathbf{u}}_3^{[G]} \right) \\
\dot{\mathbf{u}}_3^{[p]} &= \dot{\mathbf{u}}_3^{[G]} + \omega_1 \left(\mathbf{u}_2^{[p]_0} + \dot{\mathbf{u}}_2^{[p]} \Delta t / 2 - \bar{\mathbf{u}}_2^{[G]} \right) - \omega_2 \left(\mathbf{u}_1^{[p]_0} + \dot{\mathbf{u}}_1^{[p]} \Delta t / 2 - \bar{\mathbf{u}}_1^{[G]} \right)
\end{aligned} \tag{54}$$

La ecuación (54) puede ser expresada en forma matricial como

$$[\mathbf{A}]\{\dot{\mathbf{x}}_i^{[p]}\} = \{\mathbf{b}\} \tag{55}$$

donde

$$\begin{aligned}
[\mathbf{A}] &= \begin{bmatrix} 1 & \omega_3 \Delta t / 2 & -\omega_2 \Delta t / 2 \\ -\omega_3 \Delta t / 2 & 1 & \omega_1 \Delta t / 2 \\ \omega_2 \Delta t / 2 & -\omega_1 \Delta t / 2 & 1 \end{bmatrix} \\
\{\mathbf{b}\} &= \left\{ \begin{aligned} &\dot{\mathbf{u}}_1^{[G]} + \omega_2 \left(\mathbf{u}_3^{[p]_0} - \bar{\mathbf{u}}_3^{[G]} \right) - \omega_3 \left(\mathbf{u}_2^{[p]_0} - \bar{\mathbf{u}}_2^{[G]} \right) \\ &\dot{\mathbf{u}}_2^{[G]} + \omega_3 \left(\mathbf{u}_1^{[p]_0} - \bar{\mathbf{u}}_1^{[G]} \right) - \omega_1 \left(\mathbf{u}_3^{[p]_0} - \bar{\mathbf{u}}_3^{[G]} \right) \\ &\dot{\mathbf{u}}_3^{[G]} + \omega_1 \left(\mathbf{u}_2^{[p]_0} - \bar{\mathbf{u}}_2^{[G]} \right) - \omega_2 \left(\mathbf{u}_1^{[p]_0} - \bar{\mathbf{u}}_1^{[G]} \right) \end{aligned} \right\}
\end{aligned}$$

En dos dimensiones, el esquema centrado en el tiempo es usado, y la ecuación (55) se reduce a (56) y (57) como sigue:

$$\dot{\mathbf{x}}_1^{[p]} = \frac{\mathbf{P} - \beta \mathbf{Q}}{1 + \beta^2} \tag{56}$$

$$\dot{\mathbf{x}}_2^{[p]} = \frac{\mathbf{Q} + \beta \mathbf{P}}{1 + \beta^2} \tag{57}$$

donde

$$\begin{aligned}
\beta &= \frac{\omega \Delta t}{2} \\
\mathbf{P} &= \dot{\mathbf{u}}_1^{[G]} - \omega \left(\mathbf{u}_2^{[p]_0} - \bar{\mathbf{u}}_2^{[G]} \right) \\
\mathbf{Q} &= \dot{\mathbf{u}}_2^{[G]} + \omega \left(\mathbf{u}_1^{[p]_0} - \bar{\mathbf{u}}_1^{[G]} \right)
\end{aligned}$$

3.6.3 Ecuaciones aproximadas para el movimiento de un conglomerado.

El procedimiento descrito en la sección 3.6.2 involucra la solución correcta de las ecuaciones que gobiernan el movimiento. Sin embargo, este requiere el uso de un procedimiento iterativo durante

cada paso de tiempo para resolver las ecuaciones no lineales (46) para el movimiento rotacional. La velocidad computacional puede ser incrementada ignorando los términos no lineales en (46) y despreciando la relación entre las aceleraciones angulares. La aproximación es obtenida haciendo $\{\mathbf{W}\}$ y los términos no diagonales del tensor de inercia $\{\mathbf{I}\}$ iguales a cero. La aproximación solo es necesaria en casos de 3D, porque las ecuaciones gobernantes bidimensionales no contienen una no linealidad.

El procedimiento aproximado de solución debe ser apropiado para problemas casi estáticos, porque los términos diagonales del tensor de inercia pueden ser considerados como coeficientes de relajación, lo cual permite al sistema seguir un razonable (pero no el dinámico exacto) camino hacia el equilibrio. Aun para flujos estables, la aproximación debe ser razonable, porque la magnitud de las aceleraciones es pequeña; luego, la importancia de los coeficientes es pequeña. La aproximación solo debe redundar en grandes errores si la solución computada de velocidades spin es grande o si los efectos giroscópicos juegan un papel importante en el comportamiento del sistema.

3.6 - Modelo constitutivo de contacto micro estructural y evaluación de las fuerzas de contacto.

El modelo constitutivo microestructural se formula a nivel de cada contacto entre esferas o discos según sea el caso (2D y 3D). Después de haber detectado los contactos entre un par de partículas, se procede a calcular las fuerzas que ocurren en ese punto de contacto. Si se consideran dos partículas, la interacción entre estas se puede representar por la aplicación de la tercera ley de Newton. La interacción entre estos dos elementos discretos se representa por las fuerzas de contacto \mathbf{F}_1 y \mathbf{F}_2 , que por principio de acción y reacción satisfacen la relación siguiente:

$$\mathbf{F}_1 = -\mathbf{F}_2 \quad (58)$$

Si se descomponen las fuerzas (\mathbf{F}_1 o \mathbf{F}_2 que ahora se denota por \mathbf{F}) en sus componentes normal y tangencial, F_n y F_T , respectivamente:

$$\mathbf{F} = \mathbf{F}_n + \mathbf{F}_T = F_n \mathbf{n} + \mathbf{F}_T \quad (59)$$

donde:

\mathbf{n} - es el vector de la unidad normal a la superficie de la partícula en el punto de contacto (por tanto, para elementos esféricos o discos tiene la dirección de la recta que une las dos partículas y su dirección es apuntando hacia afuera respecto la partícula 1).

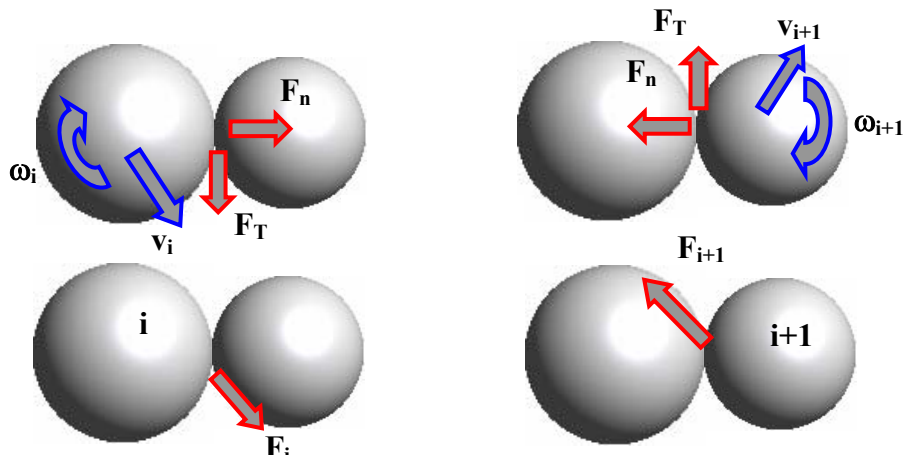


Figura 6: Descomposición de las fuerzas en sus componentes normal y tangencial.

Las fuerzas de contacto \mathbf{F}_n y \mathbf{F}_t se obtienen a través del modelo constitutivo empleado para describir el comportamiento en el contacto entre las esferas. En este caso se han formulado dos modelos constitutivos: uno con comportamiento viscoso y otro no viscoso (figura 7). El comportamiento no viscoso es un caso particular del viscoso cuando la viscosidad en el contacto es cero.

Estos modelos quedan caracterizados por diferentes elementos que describen la parte elástica y plástica del comportamiento del material y en el caso específico del modelo viscoso existe un pistón que describe el comportamiento viscoso del contacto. Simultáneamente al estudio del contacto, las fuerzas han sido descompuestas en su componente normal y tangencial; en consecuencia con esto se ha formulado el modelo constitutivo de contacto, lo que implica que la interfase del contacto se caracteriza por: una rigidez normal k_n y otra tangencial k_t , los esfuerzos máximos de cohesión en la dirección normal \mathbf{R}_n (“fuerza” máxima en la dirección normal) y tangencial \mathbf{R}_t (“fuerza” máxima en la dirección tangencial), el coeficiente de rozamiento de Coulomb μ , la intensidad de adhesión o adherencia β , y el coeficiente de amortiguamiento (c - en el modelo viscoso).

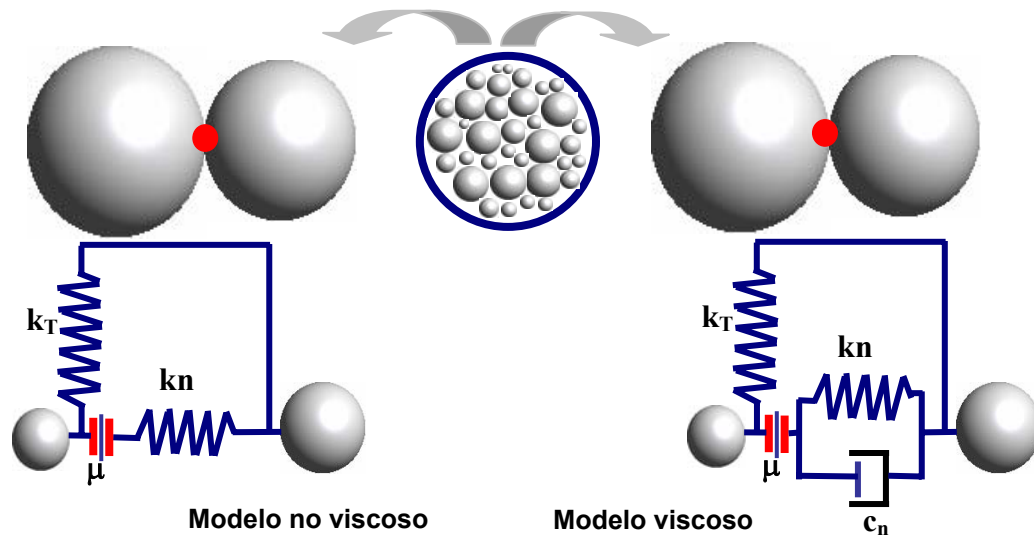


Figura 7: Tipos de modelos constitutivos de contacto.

En el caso del modelo viscoso se introduce un amortiguamiento en el contacto con la finalidad de disipar la energía cinética y disminuir las oscilaciones de las fuerzas de contacto. Por esta razón se asume la contribución de una fuerza de amortiguamiento en la componente de la fuerza de contacto normal \mathbf{F}_n en su parte elástica \mathbf{F}_{ne} y en su respectiva fuerza de contacto de amortiguamiento \mathbf{F}_{nd} . Esta fuerza \mathbf{F}_{nd} no existe en el caso del modelo no viscoso.

$$\mathbf{F}_n = \mathbf{F}_{ne} + \mathbf{F}_{nd} \quad (60)$$

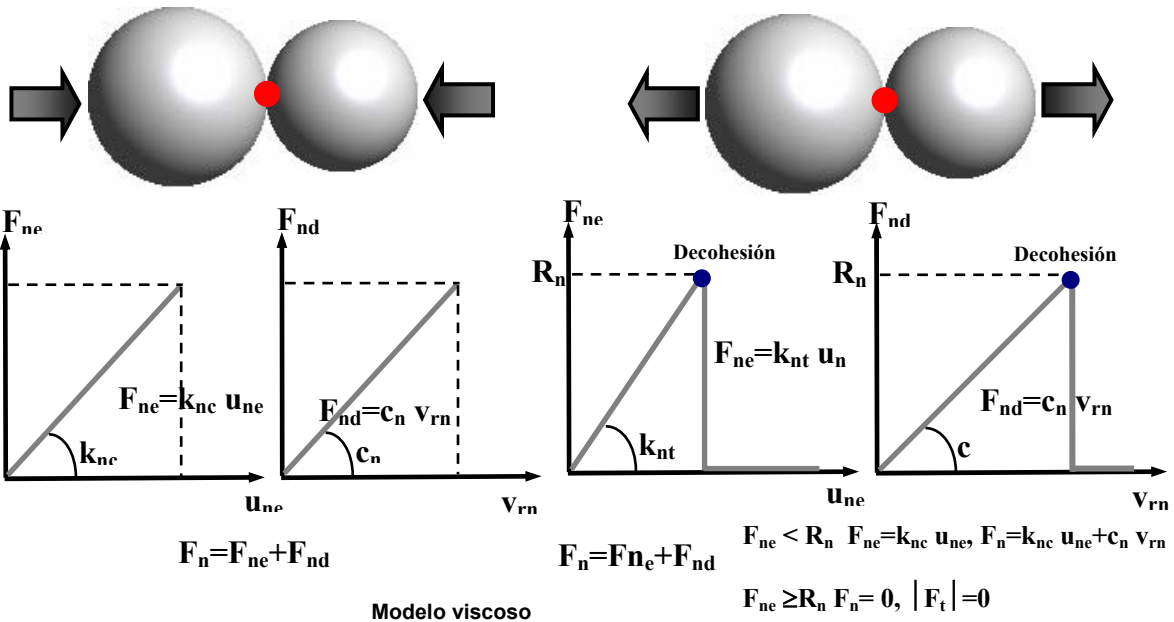
En el caso del modelo de contacto viscoso la fuerza de amortiguamiento se determina por:

$$\mathbf{F}_{nd} = \mathbf{c} \mathbf{v}_{rn} \quad (61)$$

donde la componente de fuerza de amortiguamiento es proporcional a la velocidad relativa normal \mathbf{v}_{rn} de los centros de las dos partículas en el contacto.

$$\mathbf{v}_{rn} = (\dot{\mathbf{u}}_2 - \dot{\mathbf{u}}_1) \cdot \mathbf{n} \quad (62)$$

El valor de amortiguamiento \mathbf{c} se puede tomar como la fracción del amortiguamiento crítico C_{cr} para el sistema de dos cuerpos rígidos con masas \mathbf{m}_1 y \mathbf{m}_2 , conectadas por un muelle con rigidez \mathbf{k}_n [6].



Modelo viscoso

Figura 8: Modelos constitutivos viscoso y no viscoso. Comportamiento a compresión y tracción en la dirección normal.

$$C_{cr} = 2 \sqrt{\frac{m_1 m_2 k_n}{m_1 + m_2}} \quad (63)$$

La parte elástica de la fuerza del contacto normal F_{ne} es proporcional a la rigidez normal k_n (k_{nc} – rigidez compresión y k_{nt} - tracción) y a la separación existente entre las partículas u_{rn} (figura 8).

$$F_{ne} = k_n u_{rn} \quad (64)$$

En el caso de materiales sin cohesión ($\beta=0$) no podrá desarrollarse una fuerza de contacto normal de tracción:

$$F_{ne} \leq 0 \quad (65)$$

La separación entre partículas (esferas y cilindros) u_{rn} se calcula como la distancia entre los centros (d) menos los radios (r_1 y r_2) de las esferas en contacto:

$$u_{rn} = d - r_1 - r_2 \quad (66)$$

donde:

d - distancia entre los centro de las partículas
 r_1, r_2 - radios.

Si $u_{rn} \leq 0$, la fórmula (64) es válida, y por su parte $F_{ne} = 0$.

Si el material es cohesivo ($\beta=1$) las fuerzas del contacto normales pueden ser de compresión o de tracción indistintamente. La fuerza del contacto normal debido a la cohesión es calculada por la ecuación (41), pero en este caso la separación u_{rn} es calculada como la componente relativa del desplazamiento normal entre los puntos en contacto:

$$\mathbf{u}_{rn} = \mathbf{u}_r \cdot \mathbf{n} \quad (67)$$

$$\mathbf{u}_{rn} = (\mathbf{X}_2^n + \Lambda_2^n \mathbf{r}_{c2}) - (\mathbf{X}_1^n + \Lambda_1^n \mathbf{r}_{c1}) \quad (68)$$

En el momento que se establece la adherencia cohesiva los puntos en cuestión coincide y se establece:

$$\mathbf{X}_2^0 + \Lambda_2^0 \mathbf{r}_{c2} = \mathbf{X}_1^0 + \Lambda_1^0 \mathbf{r}_{c1} \quad (69)$$

donde:

$\mathbf{X}_1^0, \mathbf{X}_2^0, \mathbf{X}_1^n$ y \mathbf{X}_2^n - denotan las posiciones del centro de las partículas en el sistema de la coordenada global en un instante de tiempo t_0 y el instante de tiempo de estudio t_n . Este aspecto denotando la configuración presente y la configuración cuando la cohesión se ha establecido.

$\Lambda_1^0, \Lambda_2^0, \Lambda_1^n$ y Λ_2^n - son las correspondientes matrices de rotación (matriz de cosenos directores),

$\mathbf{r}_{c1}, \mathbf{r}_{c2}$ - son los vectores que conectan los centros de la partícula con el punto de contacto en el momento de establecerse la conexión entre esferas.

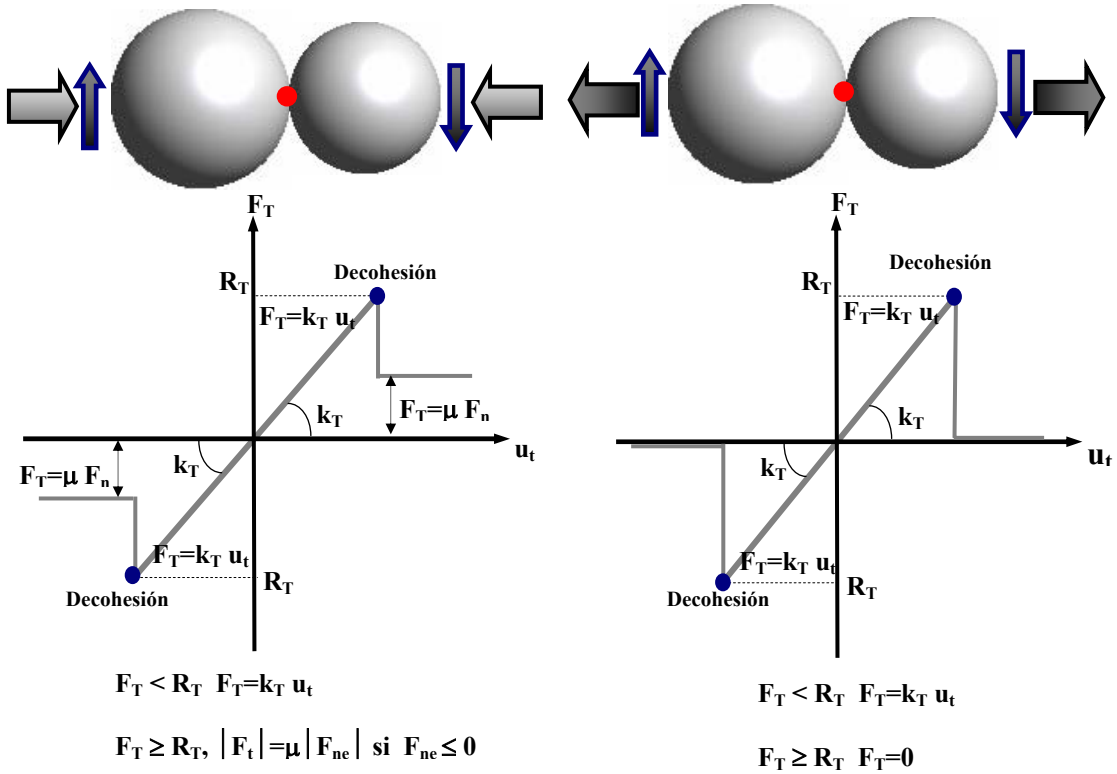


Figura 9: Modelos constitutivos viscoso y no viscoso. Comportamiento de la componente tangencial cuando en la dirección normal está sometida a compresión y a tracción.

La fuerza elástica correspondiente a la dirección tangencial (figura 9) se puede calcular por:

$$\mathbf{F}_T = \mathbf{k}_T \mathbf{u}_{rT} \quad (70)$$

donde

$$\mathbf{u}_{rT} = \mathbf{u}_r - \mathbf{u}_{rn} \mathbf{n} \quad (71)$$

La existencia de cohesión también implica que existe fuerza de contacto tangencial al mismo.

El modelo de contacto presenta un criterio de rotura microestructural que caracteriza el fallo por tracción o por cortante de los contactos entre los elementos distintos. Este aspecto queda definido por una superficie de rotura (figura 10), que se delimita por la resistencia máxima (esfuerzo máximo) a tracción (\mathbf{R}_T) y la resistencia máxima (esfuerzo máximo) a cortante (\mathbf{R}_n).

Empleando el criterio de rotura, los contactos entre las partículas pueden romperse (ruptura del aporte por cohesión entre las partículas) debido a las acciones de las cargas externas de cierto grado de magnitud. Dichos contactos se rompen cuando se exceden las fuerzas en las interfases en las direcciones normal o tangencial ($\mathbf{F}_n > \mathbf{R}_n$ o $\|\mathbf{F}_T\| > \mathbf{R}_T$). En este modelo constitutivo de contacto se considera que al romperse el contacto se ha perdido totalmente la cohesión del material ($\beta=0$). Existen otros modelos de contacto [7] que consideran la pérdida parcial de la cohesión ($0 < \beta < 1$). En la presente formulación, las ataduras cohesivas (aporte por cohesión entre las partículas), es decir, los contactos, no pueden ser restituidos después de la rotura de los mismos.

Cuando la fuerza normal de tracción alcanza un valor igual a \mathbf{R}_n , el contacto entre elementos discretos se romperá provocado por un fallo a nivel micro de tracción y dejará de existir tal fuerza. De forma análoga, cuando la fuerza tangencial alcanza un valor \mathbf{R}_T provocará un fallo de cortante microestructural y el contacto entre estos elementos discretos dejará de existir de forma análoga al caso de la falla por tracción.

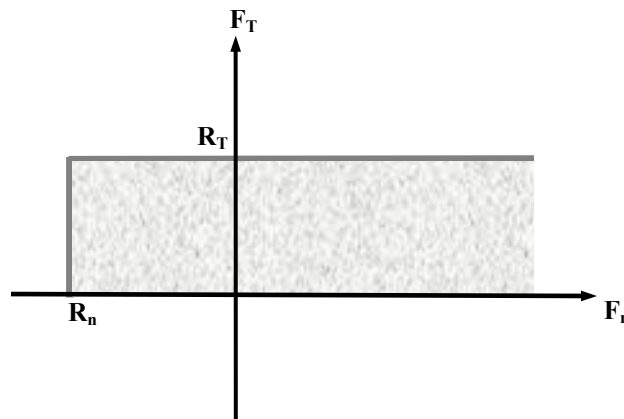


Figura 10: Superficie de rotura del modelo de contacto.

En la ausencia de cohesión o después que la cohesión entre partículas deja de existir, la reacción o fuerza tangencial puede propiciarse por el rozamiento o fricción (figura 9). Existen dos tipos de modelos básicos para simular los problemas de rozamiento o fricción entre elementos discretos: el modelo de fricción cinemática y el modelo de fricción de Coulomb. En esta formulación se hace uso del modelo de fricción de Coulomb. En el modelo de Coulomb, se produce un deslizamiento relativo entre los dos elementos distintos cuando la fuerza máxima tangencial \mathbf{R}_T alcanza cierto valor umbral que es proporcional a la fuerza de contacto. El coeficiente de proporcionalidad entre las fuerzas tangenciales y normales cuando se produce el deslizamiento (descohesión) entre las partículas o elementos discretos “*condición de slip*”, se denomina coeficiente de fricción de Coulomb. Realizando una abstracción y asumiendo que en el contacto la fuerza tangencial no alcanza el umbral, entonces no existe deslizamiento relativo entre los elementos discretos que están en contacto (“*condición stick*”).

En el modelo de Coulomb el punto clave es la evaluación de las fuerzas de rozamiento (fricción). Estas fuerzas dependen de las fuerzas de contacto normales entre elementos discretos cuando estos están en contacto. Como se relacionan esta fuerza normal de contacto es un tema complicado y complejo.

En esta formulación se pueden emplear dos tipos de modelo o leyes de comportamiento de Coulomb: Ley de Coulomb estándar o la Ley regularizada de Coulomb (figura 11).

En ausencia de cohesión o después de producirse la descohesión la fuerza tangencial se calcula por:

$$\mathbf{F}_T = -\mathbf{F}_T \frac{\mathbf{v}_{rT}}{\|\mathbf{v}_{rT}\|} \quad (72)$$

donde

\mathbf{v}_{rT} - es la velocidad tangencial relativa al punto del contacto

$$\mathbf{v}_{rT} = \mathbf{v}_r - \mathbf{v}_r \cdot \mathbf{n} \quad (73)$$

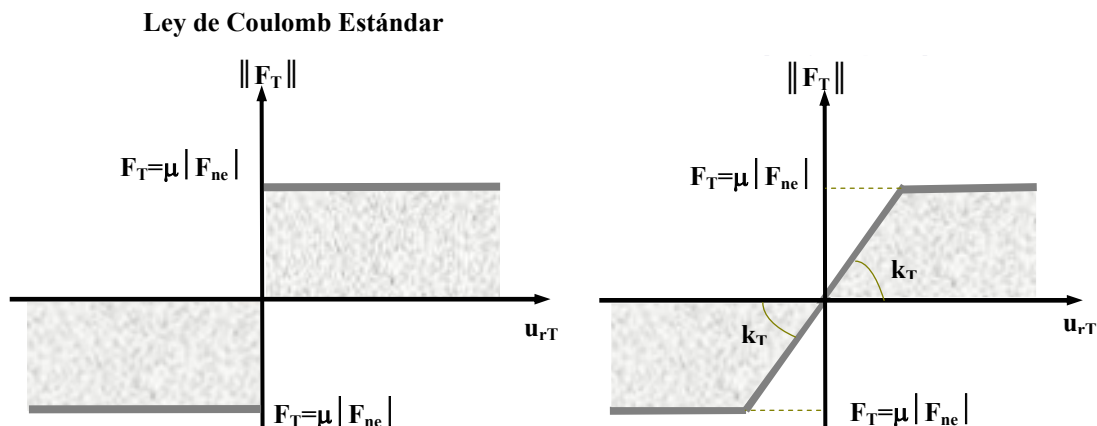


Figura 11: Leyes de Coulomb Estándar y Regularizada.

La velocidad relativa \mathbf{V}_r se obtiene de la diferencia de velocidades entre los puntos del contacto. La velocidad del punto de contacto se obtiene de la suma de la velocidad del centro de la partícula \mathbf{i} a la que pertenece y el producto de la velocidad de rotación de la misma por la distancia entre el punto de contacto y el centro.

$$\mathbf{v}_r = (\dot{\mathbf{u}}_2 + \boldsymbol{\omega}_2 \times \mathbf{r}_{c2}) - (\dot{\mathbf{u}}_1 + \boldsymbol{\omega}_1 \times \mathbf{r}_{c1}) \quad (74)$$

donde:

$\dot{\mathbf{u}}_1, \dot{\mathbf{u}}_2$ y $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2$ - son las velocidades de traslación y rotación de las partículas en contacto.

\mathbf{r}_{c1} y \mathbf{r}_{c2} - son los vectores de conectan los centros de las dos partículas en contacto.

En el modelo de Coulomb estándar se ha adoptado para calcular la fuerza de rozamiento (fricción) por:

$$\mathbf{F}_T = \mu |\mathbf{F}_n| \quad (75)$$

donde:

μ - Coeficiente de fricción de Coulomb.

Es conocido que la Ley de rozamiento (fricción) de Coulomb estándar o clásica produce una oscilación no física de la fuerza de rozamiento en la solución numérica debido a posible cambio de la dirección de la velocidad de deslizamiento. Para prevenir esta inestabilidad del modelo de rozamiento de Coulomb, dicha fuerza debe ser regularizada (figura 11).

Con la finalidad de evitar oscilaciones en la solución se limita la fuerza de rozamiento (fricción) a un valor \mathbf{F}_{Ts} . En el algoritmo numérico el modelo de Coulomb tiene que ser regulado para evitar las oscilaciones en la solución. Como consecuencia se limita el valor de la ley de rozamiento para prevenir el cambio de la dirección de la velocidad tangencial:

$$\mathbf{F}_T = \min(\mu |\mathbf{F}_T|, \mathbf{F}_{Ts}) \quad (76)$$

donde

\mathbf{F}_{Ts} , - es la llamada fuerza que retiene la fricción, es decir, la fuerza que reduciría la velocidad tangencial $\mathbf{v}_i = (\dot{\mathbf{u}}_i + \boldsymbol{\omega}_i \times \mathbf{r}_{ci})$ a cero.

Un posible proceso de regularización involucra la descomposición de la velocidad relativa tangencial en una velocidad reversible \mathbf{V}_{rT}^r e irreversible \mathbf{V}_{rT}^{ir} respectivamente.

$$\mathbf{V}_{rT} = \mathbf{V}_{rT}^r - \mathbf{V}_r^{ir} \quad (77)$$

Esto es equivalente a la formulación del problema de contacto friccional de elastoplasticidad, con lo que se puede ver claramente que la fuerza tangencial de fricción esta relacionada con el desplazamiento \mathbf{u}_{rT} . Así, primeramente se calcula:

$$\mathbf{F}_T^{\text{Pred}} = \mathbf{F}_T^{\text{Ant}} - \mathbf{k}_T \mathbf{V}_{rT} \Delta t \quad (78)$$

donde:

$\mathbf{F}_T^{\text{Pred}}$ - Fuerza de rozamiento (fricción) predicha.

$\mathbf{F}_T^{\text{Ant}}$ - Fuerza de rozamiento (fricción) calculada en el tiempo $t - \Delta t$.

y la condición de descohesión o deslizamiento se comprueba por:

$$\phi^{\text{Pred}} = \|\mathbf{F}_T^{\text{Pred}}\| - \mu \|\mathbf{F}_n\| \quad (79)$$

Si $\phi^{\text{Pred}} \leq 0$, se esta en presencia de un contacto cohesivo (“stick”) y la fuerza de rozamiento (fricción) se le asigna el valor de la fuerza de fricción predicha:

$$\mathbf{F}_T^{\text{Nuevo}} = \mathbf{F}_T^{\text{Pred}} \quad (80)$$

donde:

$\mathbf{F}_T^{\text{Nuevo}}$ - Fuerza de rozamiento (fricción) calculada en el tiempo t .

En caso de que $\phi^{\text{Pred}} > 0$, entonces:

$$\mathbf{F}_T^{\text{Nuevo}} = \mu \|\mathbf{F}_T\| \frac{\mathbf{F}_T^{\text{Pred}}}{\|\mathbf{F}_T^{\text{Pred}}\|} \quad (81)$$

3.7 - Búsqueda y Chequeo de Contacto.

3.7.1 - Generalidades al problema de Contacto.

En la simulación de problemas de ingeniería con el método de los elementos discretos, el diseño de los algoritmos de detección de contacto es de suma importancia, ya que el proceso puede ser muy exigente en cuanto a tiempo de cómputo. Generalmente, la detección de contacto se divide en dos fases diferentes, la búsqueda de vecindad y la resolución geométrica. El objetivo de la búsqueda de vecindad es identificar y listar los objetos presentes en cierta vecindad o zona alrededor del objeto analizado. La lista resultante suele llamarse la lista de vecinos. La fase de resolución geométrica compara entonces, en detalle, la geometría del elemento analizado con la de los objetos presentes en la lista de vecinos. El costo computacional depende de la complejidad y de la representación geométrica. Por ejemplo, de presentarse N caras, un esquema muy simplista requerirá un tiempo de computo del orden de $O(N^2)$ operaciones.

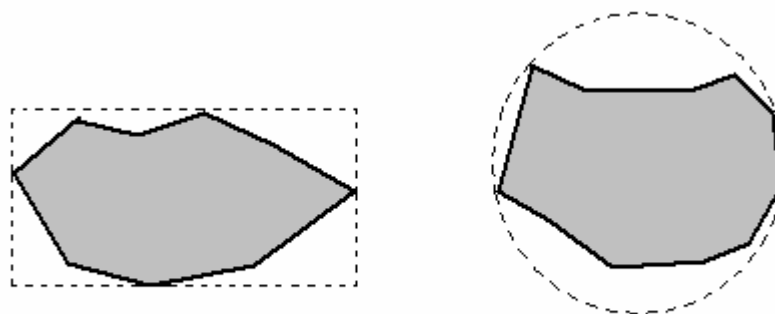
Esquemas más sofisticados tal como el DFR [O'Connor, 1996; Williams y O'Connor, 1995] requieren $O(N^{1/2})$ operaciones. De manera similar, el costo de la búsqueda de vecindad depende del número de objetos a chequear y es más potente a pequeña escala; con un esquema poco desarrollado donde se chequea cada objeto en relación con cada uno de los demás, el costo es de $O(N^2)$ operaciones.

Como casi siempre el número de elementos tratados en simulaciones de elementos discretos es elevado, se entiende que la búsqueda de vecindad puede ser un “cuello de botella” computacional.

Separando los algoritmos de búsqueda de vecindad de los de resolución geométrica, es posible considerar el problema de la búsqueda de vecindad sin referirse a los detalles de la geometría local. Esta separación, acabada mediante el uso de “volúmenes circundantes”, asegura que el algoritmo de búsqueda de vecindad quede válido para cualquier representación geométrica y esquema de resolución escogido. Esta robustez es esencial en el desarrollo de un sistema generalizado de simulación mediante el MED, donde una variedad de representaciones geométricas pueden ser usadas en la actualidad.

3.7.2 - Detección de contacto.

La interfase entre la geometría local y la búsqueda global de vecindad se lleva acabo mediante el uso de volúmenes circundantes.



Volumenes circundantes

Figura 12: Volúmenes circundantes empleados en la búsqueda de vecindad

Actuando sobre el volumen circundante en vez de la geometría local, el algoritmo de búsqueda de vecindad puede tratar cualquier representación geométrica de la misma forma simplificada. Aun cuando cualquier volumen circundante puede usarse, la esfera es el más común. La esfera se representa simplemente mediante su posición y su radio, y además es invariante frente a la rotación, así que su radio necesita ser computado una sola vez para cada elemento de la simulación.

El algoritmo de búsqueda de vecindad utiliza los volúmenes circundantes para determinar una lista de vecinos, para cada objeto. El juego completo de todas las parejas candidatas (C) determinado por el algoritmo de búsqueda de vecindad viene dado por todas los pares (blanco/candidato al contacto) sacados de las listas de vecinos, después de haber quitado los duplicados.

El algoritmo de búsqueda de vecindad hipotéticamente ideal retornaría un juego de candidatos C_{exact} que es exacto para el volumen circundante analizado en el sentido de que no incluye ningún par de elementos cuyos volúmenes circundantes no se solapan.

En la realidad, los algoritmos implementados usan esquemas simplificados que identifican rápidamente una aproximación conservadora de C_{exact} . Esta aproximación puede incluir pares cuyos volúmenes circundantes no se solapan, pero debe incluir todos los pares cuyos dichos volúmenes se solapan. Cualquier aproximación de este tipo puede ser reducida a C_{exact} aplicando simplemente una prueba exacta de intersección al volumen circundante de cada miembro. Con esta prueba empieza la fase de resolución geométrica. Esta división pone de relieve el compromiso entre la reducción del tiempo de cómputo (T) y la reducción del tamaño del juego de candidatos C . Estos dos objetivos son respectivamente la velocidad y la precisión. En la mayoría de los algoritmos implementados hasta la fecha, C es proporcional a C_{exact} (y por consiguiente a N), pero T puede ser del orden de N^2 . Esto significa que en el caso de simulaciones arbitrariamente grandes, la velocidad será de mayor consideración.

Con la aparición de representaciones geométricas más complejas, se hace necesario considerar el proceso de detección más allá de simples geometrías esféricas, y desarrollar métodos más generales.

Aun cuando el uso de volúmenes circundantes aísla efectivamente la búsqueda de vecindad de los detalles de la geometría local, y permite considerar cada cuerpo como una esfera, objetos no esféricos pueden tener volúmenes circundantes significativamente mayores que esferas de tamaño similar.

Cuando se combina esto a variaciones de tamaño de los objetos, puede llevar a una amplia distribución de tamaños de esferas circundantes. Los objetos no esféricos pueden permitir también un mayor grado de solape entre volúmenes circundantes y, así, aumentar la densidad del vecindario de contacto.

Estos tipos de heterogeneidades pueden contribuir al detrimento del rendimiento y de la precisión de los algoritmos de búsqueda de vecindad y representan consideraciones muy importantes a la hora de su diseño y uso.

3.7.3 - Problema de Contacto entre elementos discretos esféricos.

Los pares de esferas en contacto deben detectarse automáticamente durante el proceso de cálculo. Si se chequea el contacto de cada esfera contra todas las demás el tiempo de computación sería proporcional a n^2 , siendo n el número de elementos.

Se han desarrollado varios métodos para la determinación de los contactos mucho más efectivos. Previamente a la detección del contacto se suelen ordenar los elementos espacialmente mediante un algoritmo que los agrupe. De este modo, se determinan los elementos vecinos y sólo es necesario chequear el contacto entre estos.

En ausencia de cohesión, el contacto se determina cuando una esfera ha penetrado en la contigua. En el caso de esferas o cilindros simplemente se debe verificar que:

$$u_{rn} \leq 0$$

en donde u_{rn} se calcula como $u_{rn} = d - r_1 - r_2$, tal.

Si el contacto es cohesivo, la condición precedente se reemplaza por:

$$u_{rn} \leq u_{rn,max}^+$$

en donde $u_{rn,max}^+$ es la separación que debe crearse para que el contacto entre dos esferas o discos se rompa por completo, la cual debe determinarse en base al modelo constitutivo y las propiedades del material.

La determinación del contacto es sencilla, simplemente hay que verificar la ecuación correspondiente en función de si existe o no cohesión. Por tanto, nuestro problema se reduce a la efectividad del algoritmo de agrupación de los elementos. Los algoritmos más populares para llevar a cabo esta operación se pueden agrupar en:

- Subdivisión en celdas
- Árboles binarios
- Celdas centradas en el cuerpo
- Spatial heap sort y Spatial sorting.
- NBS.
- Quadtree (2D) y Octree (3D)

3.7.3.1 - Subdivisión en celdas

El espacio que engloba el problema de estudio se discretiza en celdas rectangulares (2D) o hexaédricas (3D) del mismo tamaño. Los elementos se asocian a las celdas en base a sus coordenadas. La eficacia del método depende del equilibrio entre el tamaño de la celda y el

número de elementos por celda. La eficiencia del método depende de que los elementos estén distribuidos entre las celdas. Un método adaptativo puede ser implementado en caso de que la distribución de elementos no sea uniforme.

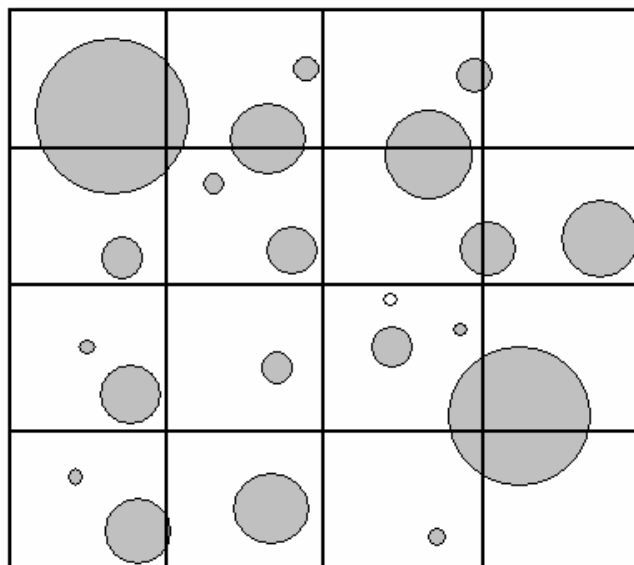


Figura 13: Subdivisión regular en celdas

3.7.3.2 - Árboles binarios

Un método efectivo de clasificación para una distribución espacial no ordenada de los elementos se basa en las estructuras en árbol binarias. El espacio se divide en celdas rectangulares que pueden contener como máximo dos elementos. En caso de que el número de elementos en una celda sea mayor que dos esta se divide en dos. La subdivisión de las celdas se realiza alternativamente en el eje x, z e y. La representación del espacio, si se ha discretizado de este modo, se realiza cómodamente mediante una estructura en árbol binaria. Dicha estructura permite

localizar fácilmente objetos pertenecientes a un subdominio recorriendo el árbol de arriba hacia abajo. El costo de dicha búsqueda es del orden de $O(N \cdot \log_2 N)$.

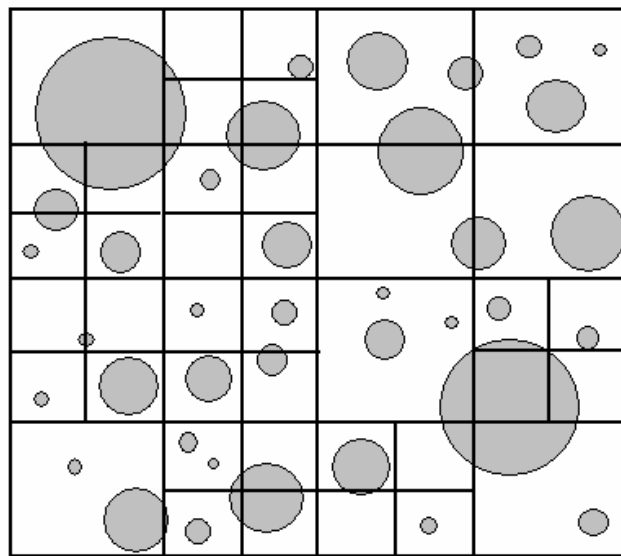


Figura 14: Subdivision adaptativa del espacio en celdas

3.7.3.3 - Celdas centradas en el cuerpo

Este método de detección del contacto se beneficia de la suposición de que la configuración de los elementos evoluciona lentamente y que, por tanto, los nuevos contactos pueden formarse únicamente entre elementos que estén lo suficientemente cercanos en el paso de estudio. La lista de contactos potenciales para cada elemento incluye los objetos que estén contenidos en una celda que englobe el elemento de estudio. Este método combinado con el de Quadtree para 2D y el de Octree para 3D es el implementado en el algoritmo de detección del contacto utilizado.

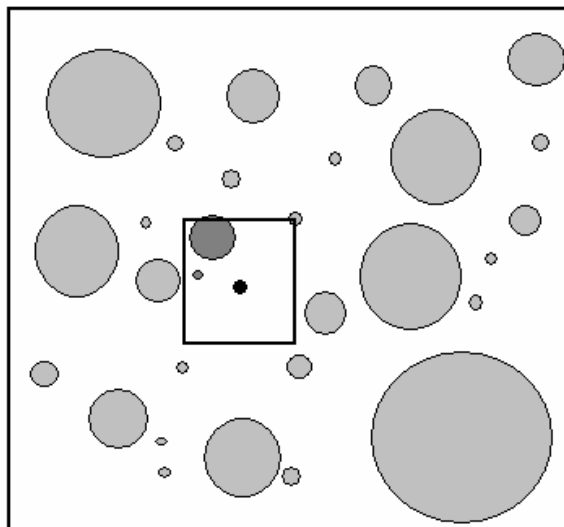


Figura 15: Celda centrada en el cuerpo.

3.7.3.4 - Spatial heap sort y Spatial sorting

Las bases del método Spatial heap sort es la clasificación de los elementos en base a sus coordenadas. Los elementos se almacenan en un árbol binario. La búsqueda de un elemento en este tipo de estructura es del orden de $O(N \cdot \log_2 N)$. Cada objeto se ve clasificado en función de su menor coordenada en cada eje. Después de definir de forma arbitraria un eje como “master” y el otro como “slave”, cada objeto (blanco) se analiza en orden a lo largo del eje “master”. Mediante una búsqueda binaria se encuentran los límites superiores del blanco. Para el eje “master”, los límites superiores e inferiores permiten delimitar un rango de objetos en la lista ordenada de dicho eje, que potencialmente intersecan el objeto blanco. Es el rango “master”.

En el caso del eje “slave”, se extiende el límite inferior hasta el objeto con la menor coordenada que pueda intersecar el blanco. El rango de objetos encontrándose entre el límite inferior extendido del eje “slave” y su límite superior se denomina rango “slave”.

La intersección de ambos rangos determina el juego final de candidatos. (figura 16)

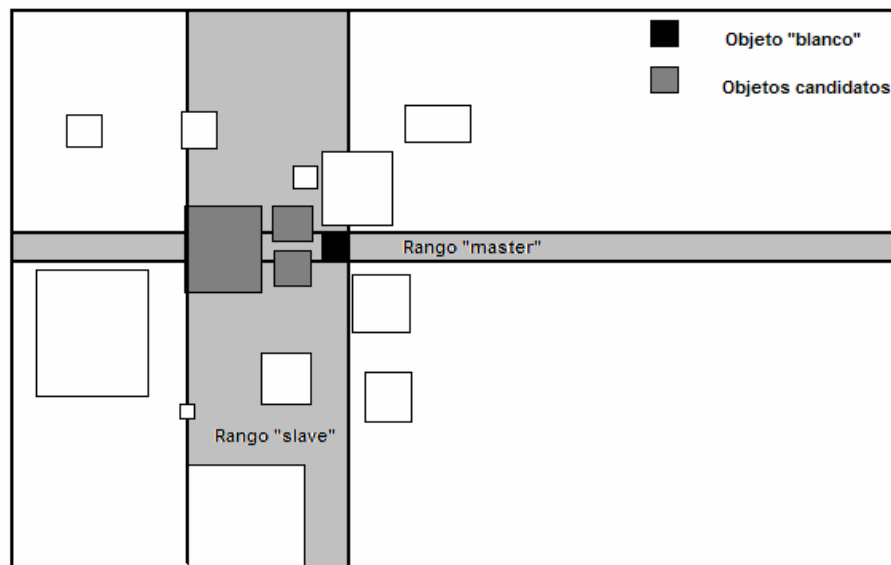


Figura 16: Delimitación de los rangos “slave” y “master” en el Spatial sorting

3.7.3.5 - NBS.

3.7.3.5.1 - Generalidades.

El algoritmo NBS (No Binary Search) es conveniente para problemas involucrando un gran número de cuerpos y desplazamientos significativos de estos. El tiene un tiempo total de detección T proporcional al número de elementos discretos, o sea $T \propto N$.

Los requerimientos de memoria total son proporcionales al número total de elementos

$$\text{discretos: } M \propto \begin{cases} N \frac{1}{\sqrt[2]{\rho}} & \text{problemas en 2D} \\ N \frac{1}{\sqrt[3]{\rho}} & \text{problemas en 3D} \end{cases}$$

donde: ρ es la densidad del medio (el volumen total de elementos discretos por unidad de volumen).

El algoritmo NBS trabaja tan bien para ensamblajes densos como sueltos, con un tiempo de cómputo independiente de la densidad del medio e incrementos insignificativos de requerimientos de memoria debido a una disminución de la densidad de ensamblajes.

3.7.3.5.2 - Definición del Problema de Detección de Contacto en el algoritmo NBS.

El algoritmo de detección de contacto NBS se basa en la suposición que cada elemento discreto se puede asimilar, para los propósitos de detección de contacto, a una esfera en 3D o a un disco en 2D. El diámetro de un disco equivalente $2r$ se obtiene a partir del elemento discreto de mayor tamaño del sistema, o sea que todos los elementos discretos son aproximados mediante discos similares. Entonces el algoritmo asume un sistema constituido de N discos idénticos ocupando un espacio

finito de forma rectangular.

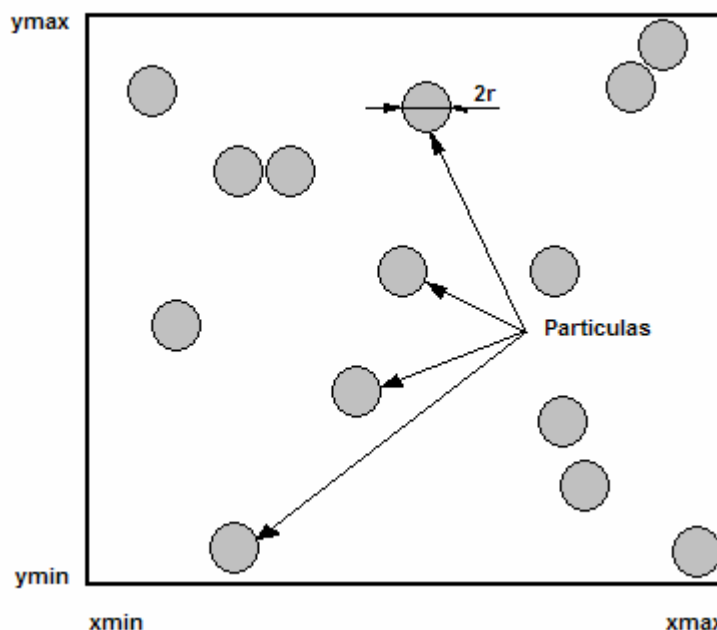


Figura 17: Espacio de análisis del algoritmo NBS

Los límites del espacio se definan como x_{\max} , x_{\min} , y_{\max} , y_{\min} . No pueden existir círculos cuyo centro tenga coordenadas mayores que x_{\max} , y_{\max} , o menor que x_{\min} , y_{\min} . El objetivo es de encontrar todos los pares de discos lo suficientemente próximos como para que la distancia entre sus puntos mas cercanos sea menor o igual a cero, en otras palabras, que se solapan o están en contacto.

3.7.3.5.3 - Descomposición del Espacio en el Algoritmo NBS.

El algoritmo NBS se basa en la descomposición del espacio. Este se divide en celdas cuadradas de tamaño $2r$.

A cada disco se le asigna un número (entero) de identificación $\{0, 1, 2, \dots, N-1\}$.

De forma similar, a cada celda se le asigna un par de números (enteros) de identificación (i_x, i_y) , donde $i_x = 0, 1, 2, 3, \dots, n_{\text{celx}} - 1$ y $i_y = 0, 1, 2, \dots, n_{\text{cely}} - 1$ con n_{celx} y n_{cely} como el número total de celdas en las direcciones X y Y respectivamente.

$$n_{\text{celx}} = \frac{x_{\max} - x_{\min}}{2r}$$

$$n_{\text{cely}} = \frac{y_{\max} - y_{\min}}{2r}$$

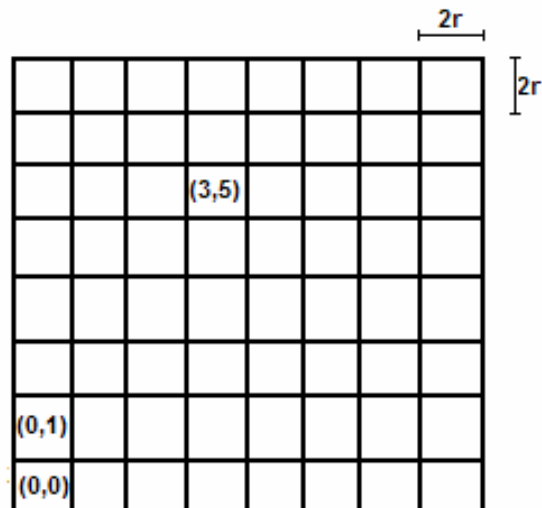


Figura 18: Espacio cuadriculado del algoritmo NBS

La proyección del juego de discos

$$\mathbf{E} = \{0, 1, 2, \dots, \mathbf{N} - 1\}$$

sobre el juego de celdas

$$\mathbf{C} = \left\{ \begin{array}{cccc} (0,0), & (0,1), & \dots & (0, n_{\text{cely}} - 1). \\ (1,0), & (1,1), & \dots & (1, n_{\text{cely}} - 1). \\ \vdots & \vdots & \ddots & \vdots \\ (n_{\text{celx}} - 1, 0), & (n_{\text{celx}} - 1, 1), & \dots & (n_{\text{celx}} - 1, n_{\text{cely}} - 1). \end{array} \right\}$$

se define de forma tal que cada elemento discreto esté asignado a una sola y única celda. El disco con coordenadas (\mathbf{x}, \mathbf{y}) se asigna a la celda $(\mathbf{i}_x, \mathbf{i}_y)$ donde

$$i_x = \text{Int} \left(\frac{\mathbf{x} - \mathbf{x}_{\min}}{2r} \right)$$

$$i_y = \text{Int} \left(\frac{\mathbf{y} - \mathbf{y}_{\min}}{2r} \right)$$

3.7.3.5.4 -- Detección de Contacto en el Algoritmo NBS.

La detección del contacto se efectúa mediante el chequeo de todos los discos proyectados en una celda particular contra los discos de las celdas vecinas. Los discos proyectados en la celda $(\mathbf{i}_x, \mathbf{i}_y)$ se chequean para el contacto con los discos proyectados en las celdas $(\mathbf{i}_x, \mathbf{i}_y)$, $(\mathbf{i}_x - 1, \mathbf{i}_y)$, $(\mathbf{i}_x - 1, \mathbf{i}_y - 1)$, $(\mathbf{i}_x, \mathbf{i}_y - 1)$, y $(\mathbf{i}_x + 1, \mathbf{i}_y - 1)$. De esta forma, los discos asignados a celdas no vacías van a chequearse contra los discos proyectados en las celdas adyacentes.

3.7.3.6 - Quadtree (2D) y Octree (3D)

Si se trabaja en 2 dimensiones el dominio se divide en celdas rectangulares con un máximo de cuatro elementos por celda. Si una celda contiene más de cuatro elementos esta se divide en cuatro celdas. Al discretizar el dominio así se puede representar mediante un árbol de cuatro

ramas (Quadtree). Dicha estructura permite identificar fácilmente objetos pertenecientes a un subdominio recorriendo el árbol de arriba hacia abajo. El costo de dicha búsqueda es del orden de $O(N \cdot \log_4 N)$.

La extrapolación de este concepto a las tres dimensiones del espacio da lugar a las estructuras de ocho elementos (Octree). Ahora el dominio se subdividirá en celdas que contengan ocho elementos como máximo, si en una celda hay más de ocho elementos ésta se subdividirá en ocho celdas. Para buscar elementos pertenecientes a un subdominio se recorre el árbol de arriba hacia abajo como en los casos anteriores y el costo de la búsqueda será del orden de $O(N \cdot \log_8 N)$.

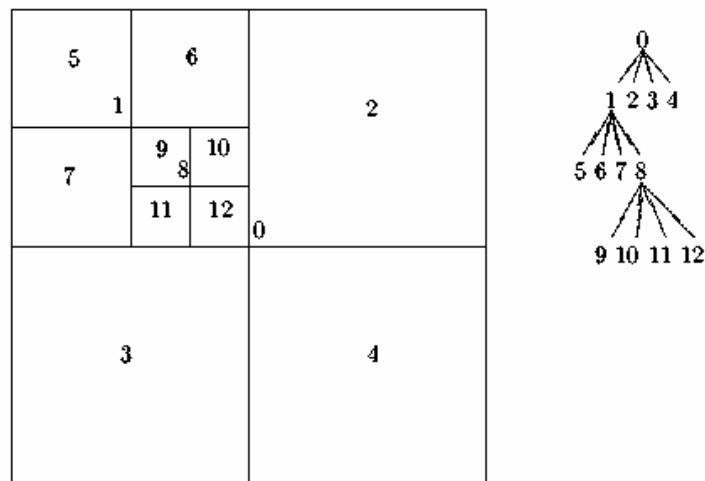


Figura 19: Estructura Quadtree

En la formulación implementada la búsqueda del contacto se basa en estructuras Quadtree en 2D y Octree en 3D combinadas con la técnica de agrupación basada en las celdas centradas en el cuerpo. De este modo el costo computacional del proceso pasa a ser proporcional a $n \cdot \ln(n)$, permitiéndose usar sistemas de mayor tamaño.

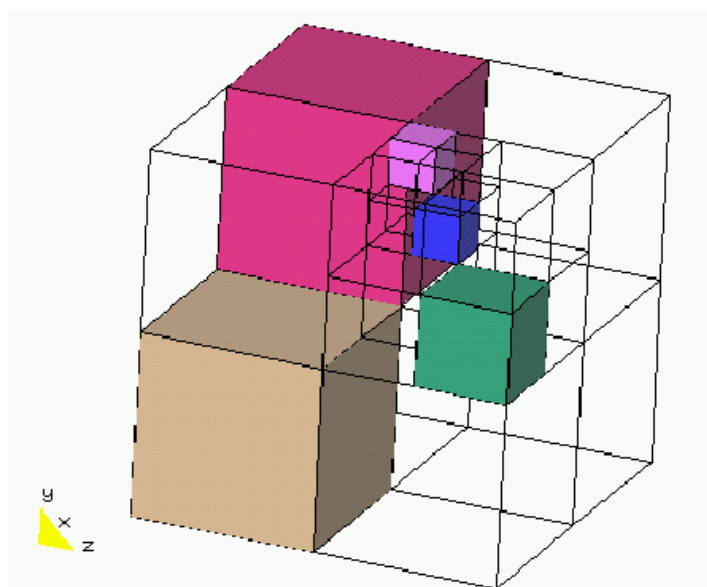


Figura 20: Estructura Octree

La construcción de estructuras Quadtree (2D) y Octree (3D) en cada paso de tiempo sería muy caro. Para pasos de tiempo muy pequeños, la mayoría de contactos pueden ser los mismos que en los pasos anteriores. La utilización de la información sobre los pares de contactos existentes en el paso de tiempo anterior puede agilizar la búsqueda de contactos en el actual. Por este motivo el algoritmo de búsqueda consta de dos etapas: una primera consistente en una búsqueda global de pares de elementos potencialmente en contacto basada en las estructuras quadtree y octree; y una búsqueda local verificando la lista de contactos potenciales basada en el método de las celdas centradas en el cuerpo.

3.7.3.6.1 - Búsqueda global

Los elementos se organizan en el espacio mediante estructuras octree y quadtree. Posteriormente se crea una lista de contactos existentes y contactos potenciales. El método de las celdas centradas en el cuerpo se usa para la lista de contactos potenciales. Cada objeto se envuelve por una celda circular o esférica y todos los objetos que estén en ella o que la intersecan se incluyen en la lista de contactos potenciales.

La organización del espacio por medio del método quadtree u octree y la actualización de la lista de contactos por medio de éste se realiza cada cierto número de pasos de tiempo.

3.7.3.6.2 - Búsqueda local

En cada paso de tiempo se realiza la búsqueda local. Se verifican las condiciones de contacto de los elementos de la lista de contactos potenciales y se encuentran los contactos actuales.

La celda centrada en el cuerpo tiene un radio igual al de la esfera o cilindro incrementado una cantidad determinada por el usuario c_{tol} . Dicho parámetro determina la longitud de la lista de contactos potenciales y el intervalo de tiempo entre búsquedas globales.

La metodología sería la siguiente:

- Se realiza una búsqueda global cuando:

$$u_{max} = \max_{\substack{i=1,D \\ n=1,N}} \{ u_i^n \} \geq c_{tol}$$

donde:

- D** es la dimensión del problema (2 para problemas bidimensionales y 3 para los tridimensionales);
- N** es el número total de elementos que constituyen el problema de estudio;

- \mathbf{U}_i es el desplazamiento acumulado de la esfera n en la dirección i del espacio desde la realización de la última búsqueda global.
-
- A la lista de posibles contactos de cada esfera se le añaden aquellas esferas que estén a una separación menor de $2 \cdot \sqrt{2} \cdot C_{tol}$ en problemas bidimensionales y de $2 \cdot \sqrt{3} \cdot C_{tol}$ en problemas tridimensionales.
- Se quitarán de la lista de posibles contactos de una esfera o elemento aquellas esferas que se separen hasta una distancia mayor de la mencionada

El algoritmo de contacto desarrollado es muy eficiente, permitiendo trabajar con grandes conjuntos de esferas. El valor de C_{tol} debe escogerse de tal modo que se halle el equilibrio entre el aumento de tiempo al realizar un mayor número de búsquedas globales cuando este decrece y el de invertir más tiempo en la búsqueda local al aumentar la lista de contactos potenciales cuando es menor. Un valor óptimo de C_{tol} se halla entre $0.1 \cdot r$ y $0.5 \cdot r$ (r es el radio medio de las esferas).

3.8 – Aplicación preliminar del método de partículas

Con el objetivo de ilustrar las potencialidades de la formulación realizada y de demostrar la eficiencia de su implementación computacional, se han realizado varias modelaciones en 3D del fenómeno de desgaste. El modelo de elementos discretos creado para este estudio se representa en la figura 21.

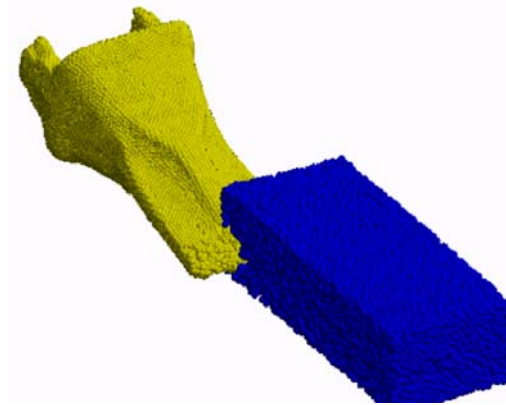


Figura 21: Modelo de elementos discretos en 3D para el estudio del desgaste.

En este estudio se ha simulado el fenómeno de desgaste de un tipo de diente en diferentes terrenos (A,B,C). Los resultados del cambio de configuración geométrica del diente producto al efecto de interacción terreno-herramienta de corte, se representa en la figura 22, 23 y 24. En la figura 22, se ilustra una vista lateral y superior del diente en diferentes intervalos de tiempo de la simulación y se observa como varía el desgaste en función del tipo terreno.

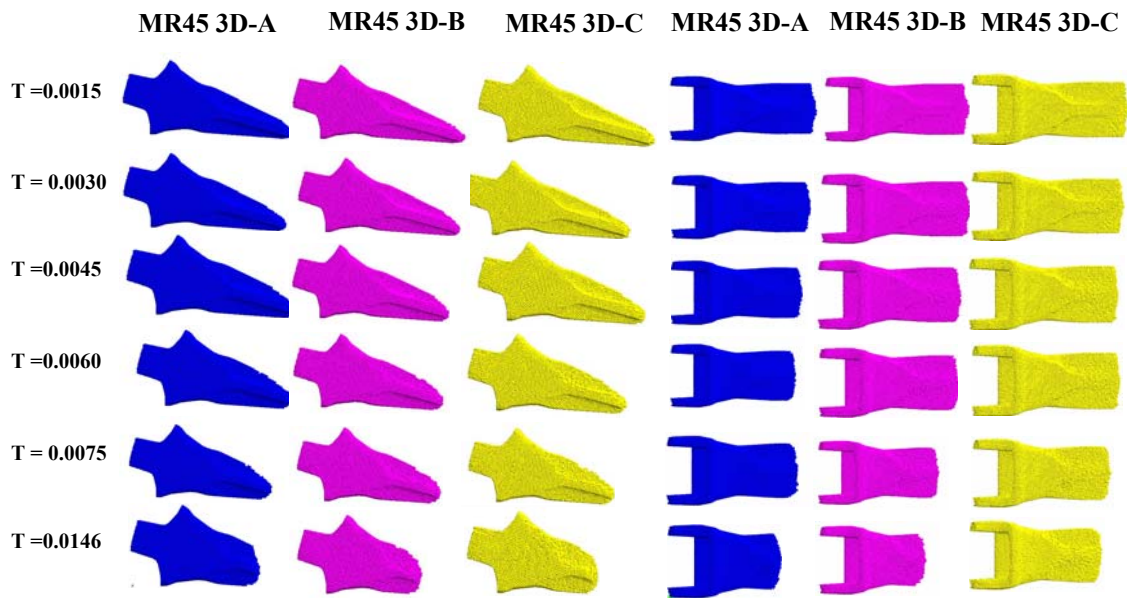


Figura 22: Estudio del fenómeno de desgaste en 3D en diferentes tipos de terreno.

En la figura 22 se evidencia el cambio de la geometría del diente cuando interactúa con el terreno (A, B y C) en dos pasos de tiempos. Se evidencia como el proceso de desgaste primeramente, se acentúa en los extremos de la herramienta de corte y después, se genera el mismo en la parte central de implemento. Estudios de este tipo posibilitan realizar análisis de configuración geométricas de diferentes herramientas de corte y además, permite realizar optimizaciones de la geometría y forma de las mismas. Estos tipos de estudios pueden estar dirigidos a la búsqueda de nuevas formas geométricas de herramientas de corte, que sean resistentes al fenómeno de desgaste. De este modo, se pueden obtener configuraciones geométricas que prorroguen la vida útil de las herramientas de corte que se emplean en las diferentes maquinarias de excavación o de otra índole. Una representación genérica del fenómeno de desgaste para un tipo de terreno se ha representado en 3 dimensiones en la figura 23 y 24.

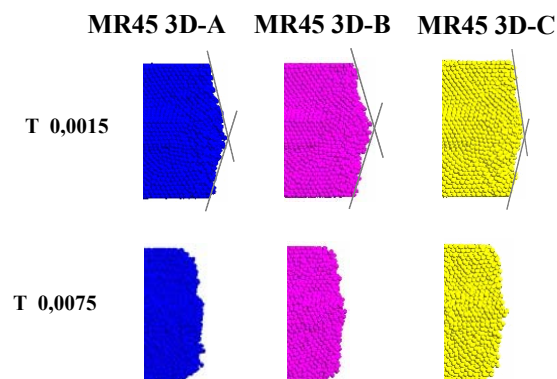


Figura 23: Evolución del fenómeno de desgaste en estadios de tiempos diferentes en diferentes tipos de terreno.

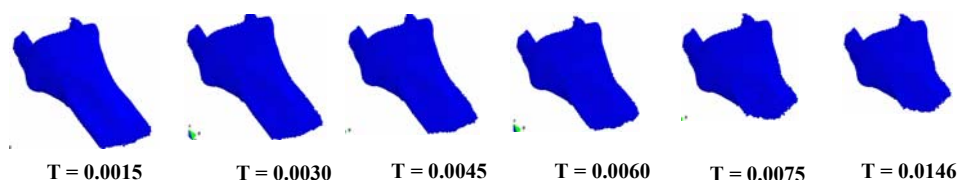


Figura 24: Representación en 3D de la evolución del desgaste de la herramienta de corte cuando interactúa con el terreno.

3.9 - Análisis de los resultados de la simulación numérica preliminar.

La finalidad de esta investigación está dirigida a demostrar la fiabilidad y confiabilidad del método de los elementos discretos en los estudios de desgaste. Con el mismo se ha demostrado la efectividad de la formulación del MED y de la eficiencia computacional realizada. El proceso de excavación puede abordarse mediante el método de los elementos discretos con resultados alentadores. Con la aplicación del MED formulado se prevé el cambio de forma de un útil a lo largo del proceso de excavación, mediante su simulación en 3D. Las simulaciones en 3D posibilitan realizar análisis del comportamiento de la geometría ante el efecto del desgaste y a su vez propicia una vía para la realización de optimizaciones de forma y geometría de los mismos. La curva de pérdida de masa con el tiempo del útil durante el proceso de excavación, obtenida mediante la simulación, proporciona una buena medida de la pérdida de masa real.

El problema de desgaste es un problema complejo. Los resultados que pueden obtenerse mediante ensayos de laboratorio son difíciles de correlacionar con situaciones de desgaste reales. Los ensayos de laboratorio y pruebas de campo ayudan a entender los mecanismos de desgaste, pero no sirven para una cuantificación directa de éste. Por tanto, la simulación del problema de desgaste mediante el método de los elementos discretos hace de nexo de unión entre los ensayos de laboratorio y la realidad. En un futuro cercano, las simulaciones de este tipo pueden resultar una herramienta muy útil para la planificación de obras y la optimización de los útiles de excavación.

3.10 – Conclusiones parciales.

Después de haber realizado la formulación del método de partículas con la finalidad de modelar problemas a nivel de la microestructura de los materiales se llega a una serie de conclusiones parciales:

- Se ha formulado el método de elementos de partículas de forma tal que se pueden emplear en dicho método alguno de los tipos de partícula trabajadas en la etapa de generación o empaquetamiento.
- Se continúa trabajando con la finalidad de incorporarle al método otros tipos de partícula ya abordadas en la etapa de generación, aspecto que no implica cambios sustanciales en la misma. De hecho dentro de semanas este trabajo será culminado.
- Con la formulación establecida se pueden resolver problemas de alto nivel de complejidad en el campo de la ingeniería con la peculiaridad de estar trabajando si se desea a escala de la micro-estructura de los materiales.

- La formulación del método permite resolver problemas multiescalas, es decir a diversas escalas de comportamiento de la mecánica computacional en la medida del tamaño de las partículas que se empleen en el proceso de simulación numérica.
- Se ha generalizado la formulación original de este método debido a haber introducido el empleo de nuevos tipos de partícula, aspecto en el que se continuará investigando para lograr incluir todos los tipos de partícula que se logran generar con la formulación de generación del medio desarrollada. Este aspecto en específico será terminado en breve tiempo ya que se dispone de los elementos y herramientas necesarias y solo se requiere formalizar la misma.

Conclusiones

Después de haber realizado una exhaustiva investigación en temas de geometría computacional (generación del medio a través de técnicas de empaquetamiento de partículas) y de mecánica computacional relacionadas con el método de partículas se ha llegado a un grupo de conclusiones. En esta dirección a modo conclusivo se pueden expresar varios aspectos de las dos temáticas:

- Se manifiesta la existencia de métodos de generación y empaquetamiento específicos para algunos tipos de partícula. No existe un método de generación o empaquetamiento de partículas de diversas formas con enfoque sistémico e integrador. En este caso se evidencia la inexistencia de un enfoque metodológico que posibilite por una misma formulación lograr empaquetar partículas de formas y posición arbitraria.
- No se evidencia en la bibliografía consultada una formulación del método de partículas que posibilite realizar simulaciones y cálculos con diversos tipos de partícula. Se manifiesta la existencia de diversas formulaciones para algunos tipos de partícula

específica y no se ha detectado la realización de una formulación más abarcadora en esta dirección.

- Los algoritmos de búsqueda de contacto presentes en la literatura consultada presentan una serie de ventajas e inconvenientes técnicos desde el punto de vista de costo computacional, por lo que es de utilidad el desarrollo de nuevas formulaciones que mejoren las deficiencias detectadas.
- Se presenta una formulación sistémica e integradora que posibilita generar o empaquetar partículas (esferas, colisiones de esferas o clusters de esferas, elipsoides, poliedros, etc.) de diversas geometrías sin necesidad de realizar transformaciones a la formulación original desarrollada. En fin, se ha presentado un algoritmo que permite generar un empaquetamiento independientemente del tipo de partículas con el que se trabaje.
- Se realiza una propuesta de un algoritmo muy eficiente para determinar si un punto se encuentra en el interior de una malla, denominado algoritmo de interioridad. Este algoritmo elimina deficiencias detectadas en otras formulaciones que tratan de determinar estas peculiaridades.
- Se han presentado nuevos métodos numéricos para resolver sistemas de ecuaciones que permiten colocar cuerpos de manera que hagan contacto con otros sin solaparlos.
- Se ha formulado el método de elementos de partículas de forma tal que se pueden emplear en dicho método alguno de los tipos de partícula trabajadas en la etapa de generación o empaquetamiento.
- Se continúa trabajando con la finalidad de incorporarle al método otros tipos de partícula ya abordadas en la etapa de generación, aspecto que no implica cambios sustanciales en la misma. De hecho dentro de semanas este trabajo será culminado.
- Con la formulación establecida se pueden resolver problemas de alto nivel de complejidad en el campo de la ingeniería con la peculiaridad de estar trabajando si se desea a escala de la micro-estructura de los materiales.
- La formulación del método permite resolver problemas multiescalas, es decir a diversas escalas de comportamiento de la mecánica computacional en la medida del tamaño de las partículas que se empleen en el proceso de simulación numérica.
- Se ha generalizado la formulación original de este método de elementos distintos debido a haber introducido el empleo de nuevos tipos de partícula, aspecto en el que se continuará investigando para lograr incluir todos los tipos de partícula que se logran generar con la formulación de generación del medio desarrollada. Este aspecto en específico será terminado en breve tiempo, ya que se dispone de los elementos y herramientas necesarias, y solo se requiere formalizar la misma.

Recomendaciones

La culminación de un trabajo científico (en este caso parcial), siempre conduce a una serie de recomendaciones que marcan pautas para futuros trabajos de investigación. Este no es la excepción a la regla, pues se ha llegado a establecer una plataforma de trabajo que permitirá resolver en el futuro muchos problemas de la física matemática y de la mecánica computacional. A partir de estos resultados y de las conclusiones enunciadas se considera oportuno realizar las siguientes recomendaciones:

- Introducir en el método de generación del medio a través de técnicas de empaquetamiento de partículas el caso de elementos geométricos que sean logrados mediante funciones de supercuádricas.
- Extender el método de partículas o de elementos distintos a todas las partículas con las cuales es posible generar el medio (partículas geométricos que sean logrados mediante funciones de supercuádricas, poliedros, elipsoides).
- Aplicar las formulaciones realizadas a la resolución de diversos problemas de la física matemática y de la mecánica computacional.

- Estudiar la complejidad y desempeño de los algoritmos propuestos.
- Realizar estudios de interrelación entre propiedades constitutivas macroestructurales y microestructurales.

Referencias Bibliográficas

Capítulo 1:

- [1] - Cundall, P.A., "A computer model for simulating progressive, large-scale movements in blocky rock systems". Symposium Soc. Internat. Mécanique des Roches, Nancy (1971), pp. 2-8.
- [2] - Chung-Yue Wangt y Vei-Chung Liangs. A packing generation scheme for the granular assemblies with planar elliptical particles. International Journal for numerical and analytical methods in geomechanics, Vol. 21, 347–358 (1997), Department of Civil Engineering, National Central University, Chungli, 32054, Taiwan, R.O.C.
- [3] - K. Nandakumar, Yiqiang Shu, and K. T. Chuang: Predicting Geometrical Properties of Random Packed Beds from Computer Simulation. Dept. of Chemical and Materials Engineering, University of Alberta, Edmonton, Alberta, Canada T6G 2G6
- [4] - Caroline Hogue. Shape representation and contact detection for discrete element simulations of arbitrary geometries. Engineering Computations, Vol. 15 No. 3, 1998, pp.

374-390. © MCB University Press, 0264-4401. Department of Engineering, University of Cambridge, Cambridge, UK

- [5] - S. Mohammadi. Discontinuum Mechanics using finite and discrete elements. Wit Press, Great Britain, (2003).
- [6] - Y. T. Feng; K. Han and D. R. J. Owen. Filling domains with disks: an advancing front approach. International journal for numerical methods in engineering Int. J. Numer. Meth. Engng 2003; 56:699–713 (DOI: 10.1002/nme.583) Department of Civil Engineering; University of Wales Swansea; Singleton Park; Swansea SA2 8PP, U.K.
- [7] - Joaquín Alberto Hernández Ortega. n. Simulación numérica de procesos de llenado mediante elementos discretos. Técnicas de mallado para medios granulares. Formulación de nuevos modelos dinámicos. Universidad Politécnica de Catalunya, E.T.S. d'Enginyeria Industrial de Terrassa. Barcelona, Directores: Dr. Juan Carlos Cante Terán, Dr. Francisco Zárate Galvá, Junio 2003
- [8] - Rainald Löhner and Eugenio Oñate. A general advancing front technique for filling space with arbitrary objects. International Journal for Numerical Methods in Engineering. 2004; 61:1977–1991. Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.10681 School of Computational Sciences, MS 4C7 George Mason University, Fairfax, VA 22030-4444, U.S.A. CIMNE, Universidad Politécnica de Catalunya Barcelona, Spain
- [9] - Xiang-Yang Li, Shang-Hua Tengz y Alper Üngör. Biting: advancing front meets sphere packing. International Journal for Numerical Methods in Engineering. 2000; 49:61-81 Department of Computer Science; University of Illinois at Urbana-Champaign; Urbana; IL 61801; U.S.A.
- [10] - Goodman., R.E., R.L. Taylor and T. Brekke, "A model for the mechanics of jointed rock". Journal of Soil Mechanics and Foundation Division ASCE, v. 94, n. SM3 ,pp. 637-658 (1968).
- [11] - Zienkiewicz, O.C., B. Best, C. Dullage and K.G. Stagg, "Analysis of non-linear problems in rock mechanics with particular reference to jointed rock systems". Proc. 2nd. Cong. ISRM, Belgrade, V3, pp. 8-14 (1971).
- [12] - Burman, B.C., "A numerical approach to the mechanics of discontinua". PhD Thesis, James Cook Univ. of N. Queensland, Townsville, Australia (1971).
- [13] - Cundall, P.A.. "Computer modelling of jointed rock masses". U.S. Army Engineer Waterways Experiment Station, Vicksburg, Mississippi, Tech. Report N-78-4 (1978).
- [14] - Wilkins, M.L., "Calculation of elastic-plastic flow". Report UCRL-7322, Lawrence Radiation Laboratory, Livermore (1969).

- [15] - Cundall P.A, Strack O.D.L., "A discrete numerical model for granular assemblies", Reporte de investigaciones, 1979.
- [16] - Cundall, P.A., "UDEC A generalized distinct element program for modelling jointed rock". European Research Office, London, tech. report contract number DAJA37-79-C-0548 (1980).
- [17] - Cundall P.A , Drescher A, Strack O.D.L., "Numerical experiments on granular assemblies; Measurements and observations", Reporte de investigaciones, 1982
- [18] - Shi, G. and Goodman R.E., "Discontinuous deformation analysis-A new method for computing stress, strain and sliding of block systems". Key Questions in Rock Mechanics, Balkema, Rotterdam, pp. 381-391(1988).
- [19] - Cundall, P. A., "Distinct element models of rock and soil structure". Analytical and Computational Methods in Engineering Rock Mechanics, pp. 129-163. London (1987).
- [20] - Vieira de Lemos, J., "A distinct model for dynamic analysis of jointed rock with application to dam foundations and fault motion". PhD thesis, University of Minnesota, U.S.A. (1987).
- [21] - Shi G. y Goodman R.E, "Discontinuous deformation analysis . A new method for computing stress, strain and sliding of block systems", Reporte de investigaciones, 1988
- [22] - Brady B. and Lorig L., "Analysis of rock reinforcement using finite difference methods". Computers and Geotechnics, 5, pp. 123-149, England (1988).
- [23] - Cundall, P. A., "Formulation of a three-dimensional distinct element model-part II. Mechanical calculations for motion and interaction of a system composed of many polyhedral blocks". Int. J. Rock Mech. Sci. & Geomech Abstr. vol. 25, No. 3, pp. 107-116, England (1988).
- [24] - Ghaboussi, L, "Fully deformable discrete element analysis using finite element approach". Computer and Geotechnics vol. 5. pp. 175-195, England (1988).
- [25] - Cundall P.A., "Computer simulations of dense sphere assemblies", Reporte de investigaciones ,1988
- [26] - Brady H., "Dynamic performance and design of underground excavations in jointed rock". Static and dynamic considerations in rock engineering, Brummer (ed.), Balkema, Rotterdam (1990).
- [27] - Homand F. y Rode N. "Block modelling of jointed cliffs". Mechanics of jointed and faulted rock, Rossmanith (ed.), Balkema, Rotterdam (1990).

- [28] - Swoboda G., Stumvoll M. and Beichuan H., "Damage tensor theory and its application to tunnelling". Mechanics of Jointed and Faulted Rock, Rossmanith (ed.), Balkema, Rotterdam (1990).
- [29] - Ishida T., "Application of distinct element analysis to three simple rock models aimed at practical application to toppling failure of fissured rock slopes", Reporte de investigaciones, 1990
- [30] - Chang C.S. and Chao S.J., "Discrete Element Analysis for Active and Passive Pressure Distribution on Retaining Wall". Computers and Geotechnics. vol. 16, n° 4, pp. 291--310 (1994).
- [31] - Pan D. and Reed B. "A coupled distinct element-finite element method for large deformation analysis of rock masses". Int. J. Rock Mech. Min. Sci. & Geomech Abstr. vol. 28, No. 1, pp. 93-99, England (1991).
- [32] - Hocking, G., "The discrete element method for analysis of fragmentation of discontinua". Engineering Computations, vol. 9, pp. 145-155 (1991).
- [33] - Chang C.S. and Chao S.J., "Discrete Element Method for Bearing Capacity Analysis". Computers and Geotechnics, vol. 12, n° 4, pp. 27-288 (1991).
- [34] - Arumugan A., Zaman M. y Laguros J., "Finite element algorithm for jointed concrete pavements subjected to moving aircraft". Computers and Geotechnics, vol. 14, n°3 pp. 121,-147 (1992).
- [35] - Chang C.S., "Discrete Element Method for Slope Stability Analysis". Journal of Geotechnical Engineering, vol. 118, n° 12, pp. 1889-1905 (1992).
- [36] - Billiaux D. and Cundall P_ "Simulation des géomatériaux par la méthode des éléments Lagrangiens". Rev. Franc. Géotech. No. 63, pp.9-21 (1993)
- [37] - Topping B.H.V. and Khan A.I., "Parallel Computation Schemes for Dynamic Relaxation". Engineering Computations, vol. 11, n° 6, pp. 513-548 (1994).
- [38] - Williams JR. and O'Connor R_ "A Linear Complexity Intersection Algorithm for Discrete Element Simulation of Arbitrary Geometries". Engineering Computations, vol. 12, n° 2, pp. 185-201 (1995).
- [39] - Murijiza A., Owen D. R. J., Bicanic N. "A Combined Finite-Discrete Element Method in Transient Dynamics of Fracturing Solids". Engineering Computations, vol. 12, n° 2, pp. 145-174 (1995).
- [40] - Krishnasamy J. and Jakiela M. J., "A method to resolve ambiguities in corner-corner interactions between polygons in the context of motion simulations". Engineering Computations, vol. 12, n° 2, pp. 135-144 (1995).

- [41] - Hassen F.H, Spinnler I. y Fine I. “A new approach for rock mass cavability modelling”, Reporte de investigaciones , 1995
- [42] - Otsu M., Mori K., Osakada K., “Determination of optimum working conditions in mushy-state upsetting of magnet using distinct element simulation”, Reporte de investigaciones , 1996
- [43] - A.P.S Selvadurai, K.Sepehr, “Two-dimensional discrete element simulations of ice-structure interaction”. Reporte de investigaciones, 1998
- [44] - Han K. , Peric D., Crook A.J.L., y Owen D.R.J, “A combined finite/discrete element simulation of shot peening processes; Part one: studies on 2D interaction laws”. Reporte de investigaciones, 2000
- [45] - P. Cleary, —“Modelling comminution devices using DEM”, Reporte de investigaciones, 2001
- [46] - L.G Bierowski, Shiro Maeno, Hitosho Gotoh, and Eiji Harada, “DEM-FEM model of the outflow of backfilling sand from behind a seawall under wave motion”. Reporte de investigaciones, 2002
- [47] - J.F. Hazzard, R.P. Young, S.J. Oates, “Numerical modelling of seismicity induced by fluid injection in a fractured reservoir”, Reporte de investigaciones, 2003
- [48] - J. F. Favier, M.H. Abbaspour-Fard, M. Kremmer and A. O. Raji. Shape representation of axi-symmetrical, non-spherical particles in discrete element simulation using multi-element model particles. Engineering Computations, Vol. 16 No. 4, 1999, pp. 467-480. © MCB University Press, 0264-4401. Department of Agricultural and Environmental Science, University of Newcastle, Newcastle upon Tyne, UK
- [49] - Itasca Consulting Group; Particle Flow Code in 2 Dimensions. Theory and Background.
- [50] - Didier Müller, Techniques informatiques efficaces pour la simulation de milieux granulaires par des méthodes d'éléments distincts, thèse no 1545 (1996) présentée au département de mathématiques, école polytechnique fédérale de lausanne pour l'obtention du grade de docteur ès sciences.
- [51] - J.A.Gilli, E.E.Alonso “Discontinuous numerical model for partially saturated soils at low saturation” Reporte de investigaciones ,1988
- [52] - B.C.Trent, L.G.Margolin “A Numerical laboratory for granular solids” Reporte de investigaciones 1992.

- [53] - Kuhn M.R “A flexible boundary for three dimensional DEM particles assemblies” Reporte de investigaciones, 1995
- [54] - Cundall P.A “PFC^{2D}, Particle flow code in 2 dimensions” 1995
- [55] - Tang-Tat Ng., Petrakis E. “Small-strain response of random arrays of spheres using Discrete Element Method” Reporte de investigaciones, 1996
- [56] - M.Otsu, K.Mori, K. Osakada “Inclusion of pressure of liquid phase in three-dimensional distinct element simulation of mushy-state forming.” Reporte de investigaciones, 1998
- [57] - Bagi K., Bojtar I., y Galos M. “A note on the role of internal structure of rocks in the deviations of compression strength results” Reporte de investigaciones, 1999
- [58] - Kuo-Neng, G.K.Chang, J.N.Meegoda “Micromechanical simulation of hot Mix asphalt” Reporte de investigaciones
- [59] - D.T Gethin, R.S Ransing, R.W Lewis, M.Dutko, A.J.L Crook “Numerical comparison of a deformable discrete element model and an equivalent continuum analysis for the compaction of ductile porous material” 2001
- [60] - R.Rojek, E. Oñate, F.Zarate, J. Miguel “El modelado de rocas, suelos, y materiales granulares usando elementos esféricos” 2001
- [61] - Oñate,,J. Rojek, F.Zarate, et al. “Computational modelling of mould manufacture for the lost foam process” 2003

Capítulo 2:

- [1] - Caroline Hogue: Shape representation and contact detection for discrete element simulations of arbitrary geometries. Engineering Computations, Vol. 15 No. 3, 1998, pp. 374-390. © MCB University Press, 0264-4401, *Department of Engineering, University of Cambridge, Cambridge, UK*, 1998
- [2] - M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf., Computational Geometry, Ed. Springer, 1998
- [3] - Didier Müller,: Techniques informatiques efficaces pour la simulation de milieux granulaires par des méthodes d'éléments distincts. Thèse no 1545, présentée au département de mathématiques, école polytechnique fédérale de lausanne pour l'obtention du grade de docteur ès sciences. 1996
- [4] - Irvin Pérez et al,: Generación de un medio con partículas de posición arbitraria. “Método de elementos discretos o de partículas”, Monografía, Editorial CIMNE. Reporte de Investigación Aula CIMNE-UCLV, Julio 2005, 2005.

Capítulo 3:

- [1] - P.A. Cundall and O.D.L. Strack. A discrete numerical method for granular assemblies. *Geotechnique*, 2, 47–65, 1979.
- [2] - P.G. Underwood. Dynamic relaxation. In T. Belytschko and T.J.R. Hughes, editors, *Computational Methods for Transient Dynamic Analysis*, Amsterdam. North-Holland, 1983.
- [3] - J. Argyris. An excursion into large rotations. *Comput. Meth. Appl. Mech. Eng.*, 32, 85–155, 1982.
- [4] - D.J. Benson and J.O. Hallquist. A simple rigid body algorithm for structural dynamics programs. *Int. J. Num. Meth. Eng.*, 12, 723–749, 1986.
- [5] - Itasca Consulting Group; *Particle Flow Code in 2 Dimensions. Theory and Background*, 1999.
- [6] - L.M. Taylor and D.S. Preece. Simulation of blasting induced rock motion. *Eng. Comput.*, 9(2), 243–252, 1992.
- [7] - M. Raous, L. Cangemi, and M. Cocu. A Consistent Model Coupling Adhesion, Frictional Unilateral Contact. *Comput. Meth. Appl. Mech. Eng.*, 177, 383–399, 1999.

Bibliografía

Capítulo 1:

- Anandarajah A., "Discrete Element Method for Simulating Behaviour of Cohesive Soil". *Journal of Geotechnical Engineering*, vol. 120, n° 9, pp. 1593-1610) (1994).
- Arumugan A., Zaman M. y Laguros J., "Finite element algorithm for jointed concrete pavements subjected to moving aircraft". *Computers and Geotechnics*, vol. 14, n°3 pp. 121,-147 (1992).
- Bagi K., Bojtar I., y Galos M. "A note on the role of internal structure of rocks in the deviations of compression strength results" *Reporte de investigaciones*, 1999
- Bierowski L.G, Shiro Maeno, Hitosho Gotoh, and Eiji Harada , "DEM-FEM model of the outflow of backfilling sand from behind a seawall under wave motion"., *Reporte de investigaciones*, 2002

- Billaux D. and Cundall P_ "Simulation des géomatériaux par la méthode des éléments Lagrangiens". Rev. Franc. Géotech. No. 63, pp.9-21 (1993)
- Brady B. and Lorig L., "Analysis of rock reinforcement using finite difference methods". Computers and Geotechnics, 5, pp. 123-149, England (1988).
- Brady H., "Dynamic performance and design of underground excavations in jointed rock". Static and dynamic considerations in rock engineering, Brummer (ed.), Balkema, Rotterdam (1990).
- Burman, B.C., "A numerical approach to the mechanics of discontinua". PhD Thesis, James Cook Univ. of N. Queensland, Townsville, Australia (1971).
- Burrell Sara, "Estudio del problema de desgaste empleando el Método de las Partículas". Trabajo de Diploma, Escuela Técnica Superior de Caminos Canales y Puertos de Barcelona, Universidad Politécnica de Cataluña, 2003. Tutor: Eugenio Oñate Ibañez de Navarra.
- Chang C.S. and Chao S.J., "Discrete Element Analysis for Active and Passive Pressure Distribution on Retaining Wall". Computers and Geotechnics. vol. 16, n° 4, pp. 291--310 (1994).
- Chang C.S. and Chao S.J., "Discrete Element Method for Bearing Capacity Analysis". Computers and Geotechnics, vol. 12, n° 4, pp. 27-)-288 (1991).
- Chang C.S., "Discrete Element Method for Slope Stability Analysis". Journal of Geotechnical Engineering, vol. 118, n° 12, pp. 1889-1905 (1992).
- Chung-Yue Wangt y Vei-Chung Liangs. A packing generation scheme for the granular assemblies with planar elliptical particles. International Journal for numerical and analytical methods in geomechanics, VOL. 21, 347–358 (1997) Department of Civil Engineering, National Central University, Chungli, 32054, Taiwan, R.O.C.
- Cleary P., "Modelling comminution devices using DEM", Reporte de investigaciones, 2001
- Cundall P.A , Drescher A, Strack O.D.L., "Numerical experiments on granular assemblies; Measurements and observations", Reporte de investigaciones, 1982
- Cundall P.A "PFC2D, Particle flow code in 2 dimensions", Reporte de investigaciones, 1995
- Cundall P.A, Strack O.D.L., "A discrete numerical model for granular assemblies", Reporte de investigaciones, 1979.

- Cundall P.A., "Computer simulations of dense sphere assemblies", Reporte de investigaciones ,1988
- Cundall, P. A., "Distinct element models of rock and soil structure". Analytical and Computational Methods in Engineering Rock Mechanics, pp. 129-163. London (1987).
- Cundall, P. A., "Formulation of a three-dimensional distinct element model-part II. Mechanical calculations for motion and interaction of a system composed of many polyhedral blocks". Int. J. Rock Mech. Sci. & Geomech Abstr. vol. 25, No. 3, pp. 107-116, England (1988).
- Cundall, P.A., "A computer model for simulating progressive, large-scale movements in blocky rock systems". Symposium Soc. Internat Mécanique des Roches, Nancy (1971), pp. 2-8.
- Cundall, P.A., "UDEC A generalized distinct element program for modelling jointed rock". European Research Office, London, tech. report contract number DAJA37-79-C-0548 (1980).
- Cundall, P.A.. "Computer modelling of jointed rock masses". U.S. Army Engineer Waterways Experiment Station, Vicksburg, Mississippi, Tech. Report N-78-4 (1978).
- Favier J.F, M.H. Abbaspour-Fard, M. Kremmer and A.O. Raji "Shape representation of axisymmetrical, non-spherical particles in discrete element simulation using multi-element model particles". Reporte de investigaciones ,1999
- Feng, Y. T.; K. Han and D. R. J. Owen. Filling domains with disks: an advancing front approach. International Journal for Numerical Methods in Engineering Int. J. Numer. Meth. Engng 2003; 56:699–713 (DOI: 10.1002/nme.583). Department of Civil Engineering; University of Wales Swansea; Singleton Park; Swansea SA2 8PP, U.K.
- Galassi , Mark y otros. GNU Scientific Library.
- Gethin D.T, R.S Ransing, R.W Lewis, M.Dutko, A.J.L Crook "Numerical comparison of a deformable discrete element model and an equivalent continuum analysis for the compaction of ductile porous material" 2001

- Gethin D.T, R.S Ransing, R.W Lewis, M.Dutko, A.J.L Crook “Numerical comparison of a deformable discrete element model and an equivalent continuum analysis for the compaction of ductile porous material”, Reporte de investigaciones, 2001
- Ghaboussi, L, "Fully deformable discrete element analysis using finite element approach". Computer and Geotechnics vol. 5. pp. 175-195, England (1988).
- Gilli J.A., E.E.Alonso “Discontinuous numerical model for partially saturated soils at low saturation” Reporte de investigaciones ,1988
- Goodman., R.E., R.L. Taylor and T. Brekke, "A model for the mechanics of jointed rock". Journal of Soil Mechanics and Foundation Division ASCE, v. 94, n. SM3 ,pp. 637-658 (1968).
- Graham G.W. Mustoe “A generalized formulation of the discrete element method”. Reporte de investigaciones ,1992
- Han K. , Peric D., Crook A.J.L., y Owen D.R.J, “A combined finite/discrete element simulation of shot peening processes; Part one: studies on 2D interaction laws”., Reporte de investigaciones, 2000
- Hassen F.H, SpinnlerI. y FineI. “A new approach for rock mass cavability modelling”, Reporte de investigaciones , 1995
- Hazzard,J.F. R.P. Young, S.J. Oates, “Numerical modelling of seismicity induced by fluid injection in a fractured reservoir”, Reporte de investigaciones, 2003
- Hernández Ortega, Joaquín Alberto. Simulación Numérica De Procesos De Llenado Mediante Elementos Discretos. Técnicas de mallado para medios granulares. Formulación de nuevos modelos dinámicos. Universidad Politécnica De Catalunya, E.T.S. d’Enginyeria Industrial de Terrassa. Directores: Dr. Juan Carlos Cante Terán, Dr. Francisco Zárate Galván. Barcelona, Junio 2003
- Hocking, G., "The discrete element method for analysis of fragmentation of discontinua". Engineering Computations, vol. 9, pp. 145-155 (1991).
- Hogue, Caroline. Shape representation and contact detection for discrete element simulations of arbitrary geometries. Engineering Computations, Vol. 15 No. 3, 1998, pp. 374-390. © MCB University Press, 0264-4401 Department of Engineering, University of Cambridge, Cambridge, UK.

- Homand F. y Rode N. "Block modelling of jointed cliffs". Mechanics of jointed and faulted rock, Rossmanith (ed.), Balkema, Rotterdam (1990).
- Huang H.. "Discrete Element Modeling of Tool-Rock Interaction". PhD thesis, University of Minnesota, (1999).
- Ishida T., "Application of distinct element analysis to three simple rock models aimed at practical application to toppling failure of fissured rock slopes", Reporte de investigaciones, 1990
- Itasca Consulting Group; Particle Flow Code in 2 Dimensions. Theory and Background. Ed. ITASCA 1999
- Krishnasamy J. and Jakiela M. J., "A method to resolve ambiguities in corner-corner interactions between polygons in the context of motion simulations". Engineering Computations, vol. 12, n° 2, pp. 135-144 (1995).
- Kuhn M.R "A flexible boundary for three dimensional DEM particles assemblies" Reporte de investigaciones, 1995
- Kuo-Neng, G.K.Chang, J.N.Meegoda "Micromechanical simulation of hot Mix asphalt" Reporte de investigaciones
- Kuwabara H., Aso T., Hironaka M. "Pile driving analysis using Distinct Element Method" Reporte de investigaciones
- Löhner, Rainald and Oñate, Eugenio. A general advancing front technique for filling space with arbitrary objects. International Journal For Numerical Methods In Engineering. Int. J. Numer. Meth. Engng 2004; 61:1977–1991 Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.10681 School of Computational Sciences, MS 4C7 George Mason University, Fairfax, VA 22030-4444, U.S.A. 2-CIMNE, Universidad Polit cnica de Catalunya Barcelona, Spain
- Mohammadi, S. Discontinuum Mechanics using finite and discrete elements. WIT Press, 2003.
- M ller, Didier. Techniques Informatiques Efficaces Pour La Simulation De Milieux Granulaires Par Des M thodes D' l ments Distincts, Th se No 1545 (1996) Pr sent e Au D partement De Math matiques,  cole Polytechnique F d rale De Lausanne Pour L'obtention Du Grade De Docteur  s Sciences.

- Munjiza A., K.R.F. Andrews "NBS Contact detection algorithm for bodies of similar size"
Reporte de investigaciones, 1998
- Munjiza A., Owen D. R. J., Bicanic N. "A Combined Finite-Discrete Element Method in Transient Dynamics of Fracturing Solids". Engineering Computations, vol. 12, n° 2, pp. 145-174 (1995).
- Nandakumar, K.; Yiqiang Shu, and K. T. Chuang. Predicting Geometrical Properties of Random Packed Beds from Computer Simulation. Dept. of Chemical and Materials Engineering, University of Alberta, Edmonton, Alberta, Canada T6G 2G6.
- Oñate, J., Rojek, F., Zarate, et al. "Computational modelling of mould manufacture for the lost foam process" 2003
- Otsu M., K. Mori, K. Osakada "Inclusion of pressure of liquid phase in three-dimensional distinct element simulation of mushy-state forming." Reporte de investigaciones, 1998
- Otsu M., Mori K., Osakada K "Determination of optimum working conditions in mushy-state upsetting of magnet using distinct element simulation", Reporte de investigaciones, 1996
- Otsu M., Mori K., Osakada K., "Determination of optimum working conditions in mushy-state upsetting of magnet using distinct element simulation", Reporte de investigaciones, 1996
- Pan D. and Reed B. "A coupled distinct element-finite element method for large deformation analysis of rock masses". Int. J. Rock Mech. Min. Sci. & Geomech Abstr. vol. 28, No. 1, pp. 93-99, England (1991).
- Perkins E., and J.R. Williams "A fast contact detection algorithm insensitive to object sizes"
Reporte de investigaciones, 2001
- Rojek R., E. Oñate, F. Zarate, J. Miguel "El modelado de rocas, suelos, y materiales granulares usando elementos esféricos" 2001
- Selvadurai A.P.S, K. Sepehr, "Two-dimensional discrete element simulations of ice-structure interaction", Reporte de investigaciones, 1998
- Shi G. y Goodman R.E, "Discontinuous deformation analysis . A new method for computing stress, strain and sliding of block systems", Reporte de investigaciones, 1988

- Shi, G. and Goodman R.E., "Discontinuous deformation analysis-A new method for computing stress, strain and sliding of block systems". Key Questions in Rock Mechanics, Balkema, Rotterdam, pp. 381-391(1988).
- Swoboda G., Stumvoll M. and Beichuan H., "Damage tensor theory and its application to tunnelling". Mechanics of Jointed and Faulted Rock, Rossmannith (ed.), Balkema, Rotterdam (1990).
- Tang-Tat Ng., Petrakis E. "Small-strain response of random arrays of spheres using Discrete Element Method" Reporte de investigaciones, 1996
- Topping B.H.V. and Khan A.I., "Parallel Computation Schemes for Dynamic Relaxation". Engineering Computations, vol. 11, n° 6, pp. 513-548 (1994).
- Trent B.C., L.G.Margolin "A Numerical laboratory for granular solids" Reporte de investigaciones 1992
- Underwood P.G.. "Dynamic relaxation". In T. Belytschko and T.J.R. Hughes, editors, "Computational Methods for Transient Dynamic Analysis", Amsterdam. North-Holland, (1983).
- Vieira de Lemos, J., "A distinct model for dynamic analysis of jointed rock with application to dam foundations and fault motion". PhD thesis, University of Minnesota, U.S.A. (1987).
- Wilkins, M.L., "Calculation of elastic-plastic flow". Report UCRL-7322, Lawrence Radiation Laboratory, Livermore (1969).
- Williams JR. and O'Connor R_ "A Linear Complexity Intersection Algorithm for Discrete Element Simulation of Arbitrary Geometries". Engineering Computations, vol. 12, n° 2, pp. 185-201 (1995).
- Xiang-Yang Li, Shang-Hua Tengz y Alper Üngör. Biting: Advancing front meets sphere packing. International Journal For Numerical Methods In Engineering. Int. J. Numer. Meth. Engng 2000; 49:61-81. Department of Computer Science; University of Illinois at Urbana-Champaign; Urbana; IL 61801; U.S.A.
- Zienkiewicz, O.C., B. Best, C. Dullage and K.G. Stagg, "Analysis of non-linear problems in rock mechanics with particular reference to jointed rock systems". Proc. 2nd. Cong. ISRM, Belgrade, V3, pp. 8-14 (1971).

Capítulo 2:

- Army Research Laboratory. Computer Generation of Statistical Distributions, Richard Saucier, ARL-TR-2168 March 2000
- Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf, Computational Geometry, Ed. Springer.
- Caroline Hogue: Shape representation and contact detection for discrete element simulations of arbitrary geometries. Engineering Computations, Vol. 15 No. 3, 1998, pp. 374-390. © MCB University Press, 0264-4401, Department of Engineering, University of Cambridge, Cambridge, UK, 1998
- Didier Müller,: Techniques informatiques efficaces pour la simulation de milieux granulaires par des méthodes d'éléments distincts. Thèse no 1545, présentée au département de mathématiques, école polytechnique fédérale de lausanne pour l'obtention du grade de docteur ès sciences. 1996
- Galassi, Mark y otros. GNU Scientific Library.
- Hogue, Caroline. Shape representation and contact detection for discrete element simulations of arbitrary geometries. Engineering Computations, Vol. 15 No. 3, 1998, pp. 374-390. © MCB University Press, 0264-4401. Department of Engineering, University of Cambridge, Cambridge, UK
- Lehmann, Charles. Geometría Analítica. Edición Revolucionaria, 1966
- Pérez, Irvin et al.; Generación de un medio con partículas de posición arbitraria. “Método de elementos discretos o de partículas”, Monografía, Editorial CIMNE. Reporte de Investigación Aula CIMNE-UCLV, Julio 2005, 2005.

Capítulo 3:

- D.J. Benson and J.O. Hallquist. A simple rigid body algorithm for structural dynamics programs. Int. J. Num. Meth. Eng., 12, 723–749, (1986).
- Itasca Consulting Group; Particle Flow Code in 2 Dimensions. Theory and Background. Ed. ITASCA, 1999.
- J. Argyris. An excursion into large rotations. Comput. Meth. Appl. Mech. Eng., 32, 85–155, (1982).
- L.M. Taylor and D.S. Preece. Simulation of blasting induced rock motion. Eng. Comput., 9(2), 243–252, (1992).

- M. Raous, L. Cangemi, and M. Cocu. A Consistent Model Coupling Adhesion, Frictional Unilateral Contact. *Comput. Meth. Appl. Mech. Eng.*, 177, 383–399, (1999).
- P.A. Cundall and O.D.L. Strack. A discrete numerical method for granular assemblies. *Geotechnique*, 2, 47–65, (1979).
- P.G. Underwood. Dynamic relaxation. In T. Belytschko and T.J.R. Hughes, editors, *Computational Methods for Transient Dynamic Analysis*, Amsterdam. North-Holland, (1983).

Apéndice I: Conceptualización General del Método de Monte Carlo.

Teniendo en cuenta que en el MED los cuerpos tienen dimensiones y posición aleatorias, se ha creído conveniente incluir este apéndice con las ideas fundamentales del Método de Montecarlo. Para profundizar en este tema, consúltese [4].

Conceptos Generales.

La generación de variables aleatorias, ocupa un lugar importante en las aplicaciones modernas de la Teoría de Probabilidades, en aquellas situaciones en que se requiera conocer el comportamiento estocástico de un sistema, por medio de simulación intensa por ordenador. El objetivo de la generación es obtener una muestra sintética (es decir, artificial) de valores de una variable, cuya función de densidad empírica se ajuste lo más fielmente posible a la dada como modelo probabilista de ella.

A partir de ahora, se entenderá por simular una variable aleatoria(v.a.) el proceso de obtener valores para dicha variable. A los métodos que se proponga para simular v.a. se les exigirá las siguientes características:

1. Exactitud: la muestra obtenida debe ajustarse a la distribución de la v.a. que se está simulando.
2. Eficiencia: Hay que intentar minimizar el número de operaciones necesarias para generar los valores de la variable.
3. Robustez: Los métodos diseñados han de servir para cualquier valor de los parámetros de la distribución de la v.a. a simular.

Generación de números que siguen una distribución pseudoaleatoria

Existen varios métodos generales para generar números aleatorios a partir de su función de distribución, éstos son: el método de inversión, método de composición y el método de aceptación-rechazo. También existen métodos particulares para tipos específicos de distribuciones.

Métodos directos.

Método de la función de distribución inversa.

El método de inversión se define por: Sea X una variable aleatoria con función de distribución F_x . Si la v.a. U sigue una distribución uniforme en el intervalo $[0, 1]$, entonces $Y = F_x^{-1}(U)$ tiene la misma distribución que X . Esta definición queda demostrada:

Si se define:

$$F_x^{-1}(y) = \inf\{x \in \mathfrak{R} | F_x(x) \geq y\} \quad 0 \leq y \leq 1$$

entonces:

$$F_y(x) = P\{Y \leq x\} = P\{F_x^{-1}(U) \leq x\} = P\{U \leq F_x(x)\} = F_x(x)$$

Por lo tanto, X e Y tienen la misma distribución.

La definición anterior indica una posibilidad para generar una muestra de variables X . Basta con generar números aleatorios y calcular sus inversos mediante F_x , obteniendo una muestra de variables independientes de la misma distribución que X .

Si es posible invertir la función de distribución acumulativa $F(x)$, el algoritmo es como sigue:

- (1) Generar $U \sim U(0, 1)$.
- (2) Devolver $X = F_x^{-1}(U)$.

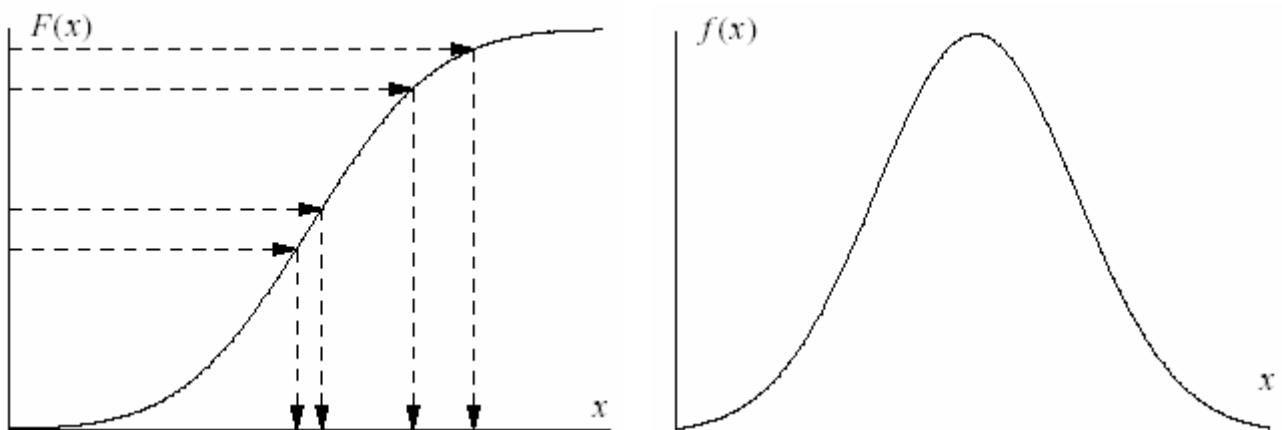


Figura 23: Método de la transformación inversa.

Composición

Esta técnica es una simple extensión de la técnica de la transformación inversa. Se aplica a situaciones donde la función de densidad de probabilidad puede ser escrita como una combinación lineal de funciones de composición más simples y donde cada una de las funciones de composición tienen una integral indefinida que es inversible. Así, se consideran los casos donde la función de densidad $f(x)$ puede ser expresada como

$f(x) = \sum_{i=1}^n p_i f_i(x)$, donde $\sum_{i=1}^n p_i = 1$ y cada una de las f_i tiene una integral indefinida F_i con una inversa conocida. El algoritmo para generar variables aleatorias mediante esta técnica es como sigue:

- (1) Seleccionar índice i con probabilidad p_i .
- (2) Independientemente generar $U \sim U(0, 1)$.
- (3) Devolver $F_i^{-1}(U)$.

Convolución

Si X y Y son variables aleatorias independientes con funciones de densidad $f_X(x)$ y $f_Y(y)$ conocidas, entonces se pueden generar nuevas distribuciones formando varias combinaciones algebraicas de X y Y . Aquí se muestra cómo esto se puede hacer mediante la suma y la multiplicación.

La técnica de convolución para la sumatoria se aplica a una situación donde la función de distribución de probabilidad puede ser escrita como una suma de otras variables aleatorias, cada una de las cuales puede ser generada directamente. El algoritmo para generar variables aleatorias mediante esta técnica es como sigue:

- (1) Generar $X_i \sim F_i^{-1}(U)$ para $i = 1, 2, \dots, n$.
- (2) Hacer $X = X_1 + X_2 + \dots + X_n$.

La técnica de convolución para la productoria se aplica a una situación donde la función de distribución de probabilidad puede ser escrita como un producto de otras variables aleatorias, cada una de las cuales puede ser generada directamente. El algoritmo para generar variables aleatorias mediante esta técnica es como sigue:

- (1) Generar $X_i \sim F_i^{-1}(U)$ para $i = 1, 2, \dots, n$.
- (2) Hacer $X = X_1 * X_2 * \dots * X_n$.

Aceptación Rechazo

Mientras que las técnicas previas son métodos directos, esta es una técnica indirecta para generar la distribución deseada. Es un método más general, el cual puede ser usado cuando métodos más directos fallan. Sin embargo, por lo general no es tan eficiente como los métodos directos. Su mayor ventaja es que siempre funcionará, aun en casos donde no hay fórmula explícita para la función de densidad (mientras haya alguna forma de evaluar la densidad en cualquier punto de su dominio).

La técnica se entiende mejor geoméricamente. Considérese una función de densidad de probabilidades arbitraria $f(x)$, mostrada en la figura 23. La motivación detrás de este método es la

simple observación de que se tiene alguna manera de generar puntos uniformemente distribuidos en dos dimensiones bajo la curva de $f(x)$, entonces la frecuencia de ocurrencia de las abscisas tendrá la distribución deseada.

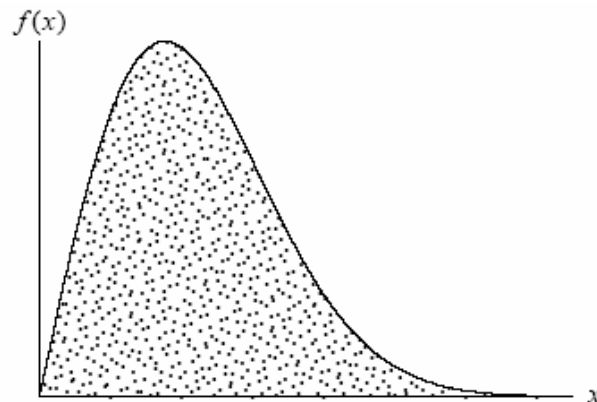


Figura 24: Densidad de Probabilidad generada de densidad de área uniforme.

Una forma simple de hacer esto es como sigue:

- (1) Seleccionar $X \sim U(0, 1)$.
- (2) Independientemente seleccionar $Y \sim U(y_{\min}, y_{\max})$.
- (3) Aceptar X si y solo si $Y \leq f(X)$.

Esto ilustra la idea, y funcionará, pero es ineficiente debido al hecho de que puede haber muchos puntos encerrados por el rectángulo delimitador que yacen sobre la función. Por eso, esto se puede hacer más eficiente encontrando primeramente una función \hat{f} que mayor a $f(x)$, en el sentido de que $\hat{f}(x) \geq f(x)$ para todo x en el dominio y al mismo tiempo, la integral de \hat{f} es inversible para todo x del dominio. Sea $\hat{F} = \int_{x_{\min}}^x \hat{f}(x)dx$ y defínase $A_{\max} = \int_{x_{\min}}^{x_{\max}} \hat{f}(x)dx$. Entonces un algoritmo más eficiente es como sigue:

- (1) Seleccionar $A \sim U(0, A_{\max})$.
- (2) Computar $X = \hat{F}^{-1}(A)$
- (3) Independientemente seleccionar $Y \sim U(0, \hat{f}(X))$
- (4) Aceptar X si y solo si $Y \leq f(X)$.

Métodos Mixtos

Este método es una combinación del método de la función inversa y el método de aceptación-rechazo.

Imagínese que la función de distribución de probabilidad es muy difícil de integrar e invertir, haciendo los métodos de la función inversa y el de aceptación-rechazo prácticamente imposibles

de aplicar. Sin embargo, supóngase que la función de distribución puede ser factorizada como sigue:

$$F(x) = f(x) g(x),$$

siendo su dominio el intervalo $[a, b]$, donde $f(x)$ es una función inversible que contiene la mayor parte de la dificultad y que $g(x)$ es relativamente cómoda pero contiene la mayor parte de la complejidad matemática. Un algoritmo para obtener valores de una variable con función de distribución $F(x)$ es:

- (1) Normalizar $f(x)$ produciendo $\tilde{f}(x)$ de manera tal que $\int_a^b \tilde{f}(x) dx = 1$.
- (2) Normalizar $g(x)$ produciendo $\tilde{g}(x)$ de manera que $\tilde{g}(x) \leq 1 \forall x \in [a, b]$.
- (3) Usando el método de la función inversa, obtener un valor x usando $\tilde{f}(x)$ como la función de distribución.
- (4) Usando esta x , aplicar el método de aceptación-rechazo usando $\tilde{g}(x)$. Es decir, escójase un número r obtenido de una distribución $U[0, 1]$. Si $\tilde{g}(x) \leq r$, aceptar x , de lo contrario, ir al paso 3.

Apéndice II: Implementaciones.

Programa de generación 2D con discos.

Constantes

```
puntos = 0.01*{{0, 0}, {14, 5}, {29, -1}, {22, 19}, {15, 15}, {10, 19},
{16, 7}, {11, 10}, {14, 10}, {14, 13}, {18, 12}};
segmentos = {{1, 2}, {2, 3}, {3, 4}, {4, 5}, {5, 6}, {6, 1}, {7, 8},
{8, 9}, {9, 10}, {10, 11}, {11, 7}};
X = Table[puntos[[i,1]], {i, Length[puntos]}];
Y = Table[puntos[[i,2]], {i, Length[puntos]}];
margenX = 0.05*(Max[X] - Min[X]); margenY = 0.05*(Max[Y] - Min[Y]);
Xmin = Min[X] - margenX; Xmax = Max[X] + margenX; Ymin = Min[Y] -
margenY;
Ymax = Max[Y] + margenY; refMallaIntX = 0.2; refMallaIntY = 0.2;
N1 = Round[1/refMallaIntX]; N2 = Round[1/refMallaIntY];
hxInt = (Xmax - Xmin)/N1; hyInt = (Ymax - Ymin)/N2;
radMin = 0.1*Min[margenX, margenY]; radMax = 6*radMin;
puntoInicial = {(Xmin + Xmax)/2, (Ymin + Ymax)/2};
puntoInicial = {0.07, 0.07}; hxBC = 2*radMax; hyBC = hxBC;
N1BC = Round[(Xmax - Xmin)/hxBC]; N2BC = Round[(Ymax - Ymin)/hyBC];
hxBC = (Xmax - Xmin)/N1BC; hyBC = (Ymax - Ymin)/N2BC; NxMalla = 50;
NyMalla = 50; hxmalla = (Xmax - Xmin)/NxMalla;
hymalla = (Ymax - Ymin)/NyMalla;
CeldasBC = Table[{}, {i, N2BC}, {j, N1BC}];
```

Mallas

```

malla = Table[N[{Xmin + i*hxmalla, Ymin + j*hymalla}], {i, NxMalla - 1},
{j, NyMalla - 1}];
malla = Partition[Flatten[malla], 2];
malla = Select[malla, Dentro[#1] & ];
mallaGraf = Graphics[Table[Point[malla[[i]]], {i, Length[malla]}]];

```

Cálculo de la estructura de interioridad

```

t0 = SessionTime[]; PuntosSalientes = {}; For[i = 1, i <=
Length[puntos],
i++, sEnt = 0; sSal = 0; For[j = 1, j <= Length[segmentos], j++,
If[segmentos[[j,1]] == i, sSal = j]; If[segmentos[[j,2]] == i,
sEnt = j]; If[sEnt > 0 && sSal > 0, Break[]];
orP = {puntos[[segmentos[[sSal,2]]]] -
puntos[[segmentos[[sSal,1]]]],
puntos[[segmentos[[sEnt,2]]]] - puntos[[segmentos[[sEnt,1]]]]};
orP = Cross[Join[orP[[1]], {0}], Join[orP[[2]], {0}]];
orP = orP[[3]] < 0; AppendTo[PuntosSalientes, orP];
CeldasInt = {}; For[i = 1, i <= N2, i++, For[j = 1, j <= N1, j++,
list = {}; Print[i, j]; For[k = 1, k <= Length[segmentos], k++,
AppendTo[list, {k, MinCellSegm[i, j, k], MaxCellSegm[i, j, k],
CortaCellSegm[i, j, k]}]; ]; list = Reducir[list];
Print[list];
list = Table[{First[list[[k1]]], Last[list[[k1]]], {k1,
Length[list]}}];
If[Count[list, {_, False}] == Length[list],
listAux = Table[First[list[[k1]]], {k1, Length[list]};
list = DentroCellwrtSeg[i, j, listAux]; AppendTo[CeldasInt,
list]; ]; CeldasInt = Partition[CeldasInt, N2];

```

Funciones

```

radios0 = Table[Random[Real, {radMin, radMax}], {100}];
radios = radios0; rA1 := Module[{sol}, sol = First[radios];
radios = Rest[radios]; sol];
DPSEgm[p_, i_] := Module[{s = {puntos[[segmentos[[i,1]]]],
puntos[[segmentos[[i,2]]]]}, p1, p2, n, sol = 0},
n = s[[2]] - s[[1]]; p1 = p - s[[1]]; p2 = p - s[[2]];
Which[p1 . n < 0, sol = Norm[p1], p2 . n > 0, sol = Norm[p2],
True,
sol = Norm[p1 - (p1 . n/Norm[n]^2)*n]]; sol = N[sol]; sol];
Reducir[l1_] := Module[{l = l1, i, j, sol},
For[i = 1, i <= Length[l] - 1, i++, For[j = i + 1, j <= Length[l],
j++,
If[l[[i,3]] <= l[[j,2]], l = Delete[l, j]; j = j - 1; ];
If[l[[j,3]] <= l[[i,2]], l = Delete[Insert[l, l[[j]], i], i +
1];
l = Delete[l, j]; j = i; ]]; sol = l; sol];
IntEstSegm[seg1_, seg2_] := Module[{n1, n2, sol}, n1 = seg1[[2]] -
seg1[[1]];
n1 = {-n1[[2]], n1[[1]]}; n2 = seg2[[2]] - seg2[[1]];

```

```

n2 = {-n2[[2]], n2[[1]]};
sol = (seg2[[1]] - seg1[[1]]) . n1*(seg2[[2]] - seg1[[1]]) . n1 <
0 &&
(seg1[[1]] - seg2[[1]]) . n2*(seg1[[2]] - seg2[[1]]) . n2 < 0;
sol];
NormCell[p_] := 2*Max[Abs[p[[1]]/hxInt], Abs[p[[2]]/hyInt]];
MinCellSegm[i_, j_, k_] := Module[{xcmin = Xmin + hxInt*(j - 1),
xcmax = Xmin + hxInt*j, ycmin = Ymax - hyInt*i,
ycmax = Ymax - hyInt*(i - 1), x, y, s, t, p, sol},
Clear[x, y]; s = {puntos[[segmentos[[k,1]]]],
puntos[[segmentos[[k,2]]]]}; p = {x, y};
sol = NMinimize[{(s[[1]] + t*(s[[2]] - s[[1]]) - p) .
(s[[1]] + t*(s[[2]] - s[[1]]) - p), xcmin <= x <= xcmax &&
ycmin <= y <= ycmax && 0 <= t <= 1}, {x, y, t}];
sol = {sol[[1]], {x, y} /. sol[[2]]}; sol];
MaxCellSegm[i_, j_, k_] := Module[{xcmin = Xmin + hxInt*(j - 1),
xcmax = Xmin + hxInt*j, ycmin = Ymax - hyInt*i,
ycmax = Ymax - hyInt*(i - 1), c1, c2, c3, c4, x, y, s, p, p1, p2,
n,
CondExt1, CondExt2, CondSeg, maxseg, max1, max2, sol},
Clear[x, y]; c1 = {xcmin, ycmin}; c2 = {xcmax, ycmin};
c3 = {xcmax, ycmax}; c4 = {xcmin, ycmax}; centro = 0.5*(c1 + c3);
s = {puntos[[segmentos[[k,1]]]], puntos[[segmentos[[k,2]]]]};
p = {x, y}; p1 = p - s[[1]]; p2 = p - s[[2]]; n = s[[2]] - s[[1]];
CondExt1 = (c1 - s[[1]]) . n <= 0 || (c2 - s[[1]]) . n <= 0 ||
(c3 - s[[1]]) . n <= 0 || (c4 - s[[1]]) . n <= 0;
CondExt2 = (c1 - s[[2]]) . n >= 0 || (c2 - s[[2]]) . n >= 0 ||
(c3 - s[[2]]) . n >= 0 || (c4 - s[[2]]) . n >= 0;
CondSeg = (c1 - s[[1]]) . n >= 0 && (c1 - s[[2]]) . n <= 0 ||
(c2 - s[[1]]) . n >= 0 && (c2 - s[[2]]) . n <= 0 ||
(c3 - s[[1]]) . n >= 0 && (c3 - s[[2]]) . n <= 0 ||
(c4 - s[[1]]) . n >= 0 && (c4 - s[[2]]) . n <= 0;
If[CondSeg, maxseg = NMaximize[{(p1 - (p1 . n/Norm[n]^2)*n) .
(p1 - (p1 . n/Norm[n]^2)*n), xcmin <= x <= xcmax &&
ycmin <= y <= ycmax && (p - s[[1]]) . n >= 0 &&
(p - s[[2]]) . n <= 0}, {x, y}], maxseg = {0, {x -> 0, y ->
0}}];
maxseg = {maxseg[[1]], {x, y} /. maxseg[[2]]};
If[CondExt1, max1 = NMaximize[{(p - s[[1]]) . (p - s[[1]]),
xcmin <= x <= xcmax && ycmin <= y <= ycmax && (p - s[[1]]) . n
<=
0}, {x, y}], max1 = {0, {x -> 0, y -> 0}}];
max1 = {max1[[1]], {x, y} /. max1[[2]]};
If[CondExt2, max2 = NMaximize[{(p - s[[2]]) . (p - s[[2]]),
xcmin <= x <= xcmax && ycmin <= y <= ycmax && (p - s[[2]]) . n
>=
0}, {x, y}], max2 = {0, {x -> 0, y -> 0}}];
max2 = {max2[[1]], {x, y} /. max2[[2]]};
sol = Max[maxseg[[1]], max1[[1]], max2[[1]]]; sol];
CortaCellSegm[i_, j_, k_] :=
Module[{s = puntos[[segmentos[[k,1]]]],
puntos[[segmentos[[k,2]]]]}, t,

```

```

centro, sol}, centro = {Xmin + hxInt*(j - 0.5), Ymax - hyInt*(i -
0.5)};
sol = Minimize[{NormCell[s[[1]] + t*(s[[2]] - s[[1]]) - centro],
0 <= t <= 1}, {t}][[1]]; sol = sol <= 1; sol];
DentroPointwrtSeg[p_, k_] := Module[{n, n12, p1, p2, sol},
p1 = p - puntos[[segmentos[[k,1]]]]; p2 = p -
puntos[[segmentos[[k,2]]]];
n12 = p1 - p2; n = {-n12[[2]], n12[[1]]}; Which[p1 . n12 < 0,
sol = PuntosSalientes[[segmentos[[k,1]]]]; sol = !sol, p2 . n12
> 0,
sol = PuntosSalientes[[segmentos[[k,2]]]]; sol = !sol, True,
sol = p1 . n >= 0]; sol];
DentroCellwrtSeg[i_, j_, listSeg_] :=
Module[{c1 = {Xmin + hxInt*(j - 1), Ymax - hyInt*i}, distancias, k,
sol},
distancias = Table[DPSegm[c1, listSeg[[k]]], {k, Length[listSeg]}];
sol = Min[distancias]; sol = First[Flatten[Position[distancias,
sol]]];
sol = listSeg[[sol]]; sol = DentroPointwrtSeg[c1, sol]; sol];
Dentro[p_] := Module[{distancias, listSeg, i, j, k, sol},
j = Quotient[p[[1]] - Xmin, hxInt] + 1;
i = N2 - Quotient[p[[2]] - Ymin, hyInt]; listSeg =
CeldasInt[[i,j]];
If[Length[listSeg] > 0, listSeg = Table[First[listSeg[[k]]],
{k, Length[listSeg]}]; distancias = Table[DPSegm[p,
listSeg[[k]]],
{k, Length[listSeg]}]; sol = Min[distancias];
sol = First[Flatten[Position[distancias, sol]]]; sol =
listSeg[[sol]];
sol = DentroPointwrtSeg[p, sol], sol = listSeg]; sol];
rAl := Random[Real, {radMin, radMax}]; CondNec[i_, j_, r_] :=
(Most[Esferas[[i]]] - Most[Esferas[[j]]]) . (Most[Esferas[[i]]] -
Most[Esferas[[j]]]) <= (2*r + Last[Esferas[[i]]] +
Last[Esferas[[j]]])^2;
Centro[i_, j_, r_] := Module[{c1, c2, r1, r2, x, y, p, sol =
{}},
Clear[x, y, p]; p = {x, y}; If[CondNec[i, j, r], c1 =
Most[Esferas[[i]]];
r1 = Esferas[[i,3]]; c2 = Most[Esferas[[j]]]; r2 =
Esferas[[j,3]];
sol = {x, y} /. NSolve[{(p - c1) . (p - c1) == (r + r1)^2,
(p - c2) . (p - c2) == (r + r2)^2}]]]; sol];
ActCeldasBC[i1_] := Module[{p, i, j}, p = Most[Esferas[[i1]]];
j = Quotient[p[[1]] - Xmin, hxBC] + 1;
i = N2BC - Quotient[p[[2]] - Ymin, hyBC];
CeldasBC = Insert[CeldasBC, i1, {i, j, 1}]; ];
Vecinas[i1_, d_] := Module[{p, i, j, i2, j2, sol},
p = Most[Esferas[[i1]]]; j = Quotient[p[[1]] - Xmin, hxBC] + 1;
i = N2BC - Quotient[p[[2]] - Ymin, hyBC];
sol = Table[CeldasBC[[i2,j2]], {i2, Max[1, i - d], Min[N2BC, i +
d]},
{j2, Max[1, j - d], Min[N1BC, j + d]}]; sol = Flatten[sol];

```

```

    sol = DeleteCases[sol, i1]; sol];
Corta[esf1_, esf2_] := Module[{c1 = Most[esf1], c2 = Most[esf2],
    r1 = Last[esf1], r2 = Last[esf2], sol},
    sol = (c2 - c1) . (c2 - c1) < (r1 + r2)^2; sol];
Admisible[esf_, ind1_, ind2_] := Module[{p, r, vec, i, j, i2, j2, cont
= 1,
    sol}, p = Most[esf]; r = Last[esf]; j = Quotient[p[[1]] - Xmin,
hxBC] +
    1; i = N2BC - Quotient[p[[2]] - Ymin, hyBC]; sol = Dentro[p];
    If[sol, vec = Table[CeldasBC[[i2,j2]], {i2, Max[1, i - 1],
        Min[N2BC, i + 1]}, {j2, Max[1, j - 1], Min[N1BC, j + 1]}];
    vec = Flatten[vec]; vec = DeleteCases[vec, ind1];
    vec = DeleteCases[vec, ind2]; While[cont <= Length[vec] && sol,
        sol = sol && !Corta[esf, Esferas[[vec[[cont]]]]];
        cont = cont + 1; ]; sol]; BorrarOutput :=
Module[{nb}, nb = SelectedNotebook[]; SelectionMove[nb, Next,
    GeneratedCell]; SelectionMove[nb, All, Cell]; NotebookDelete[nb];
];

```

Graficación

```

PlotPuntos := ListPlot[puntos, PlotStyle -> PointSize[0.02],
    DisplayFunction -> Identity];
PlotSegm := Graphics[Table[{Thickness[0.006],
    Line[{puntos[[segmentos[[i,1]]]], puntos[[segmentos[[i,2]]]]}],
    {i, Length[segmentos]}]];
PlotEtiquetas := Module[{i, sol}, sol = {}; For[i = 1, i <=
Length[puntos],
    i++, AppendTo[sol, Text[ToString[i], puntos[[i]], {0, -1}]]];
    For[i = 1, i <= Length[segmentos], i++, AppendTo[sol,
        Text[ToString[i], (puntos[[segmentos[[i,1]]]] +
            puntos[[segmentos[[i,2]]]])/2, {0, -1}]]];
    For[i = 1, i <= Length[Esferas], i++, AppendTo[sol,
        Text[ToString[i], Most[Esferas[[i]]]]]; sol = Graphics[sol];
sol];
PlotCeldasInt := Graphics[
    Join[Table[Line[{{Xmin + i*hxInt, Ymin}, {Xmin + i*hxInt, Ymax}}],
        {i, 0, N1}], Table[Line[{{Xmin, Ymin + i*hyInt},
            {Xmax, Ymin + i*hyInt}}], {i, 0, N2}]]];
PlotCeldasBC := Graphics[
    Join[Table[Line[{{Xmin + i*hxBC, Ymin}, {Xmin + i*hxBC, Ymax}}],
        {i, 0, N1BC}], Table[Line[{{Xmin, Ymin + i*hyBC},
            {Xmax, Ymin + i*hyBC}}], {i, 0, N2BC}]]];
PlotEsferas := Module[{i, sol},
    sol = Table[Circle[Most[Esferas[[i]]], Esferas[[i,3]]],
        {i, Length[Esferas]}]; sol = Graphics[sol]; sol];
GrafCentrosBC[i_, j_] := Module[{k, sol = {}},
    For[k = 1, k <= Length[Esferas], k++, If[Count[CeldasBC[[i,j]], k]
> 0,
        AppendTo[sol, k]]]; sol = Table[{Hue[1], PointSize[0.02],
        Point[Most[Esferas[[sol[[k]]]]]], {k, Length[sol]}];
    sol = Graphics[sol]; sol];
GrafTodo := Show[PlotPuntos, PlotSegm, PlotEtiquetas, PlotEsferas,
    PlotCeldasInt, AspectRatio -> Automatic, DisplayFunction ->

```

```

$DisplayFunction, Axes -> False, PlotRange -> All];
Programa Principal 1 (usando cada pivote una sola vez)
r1 = rAl; r2 = rAl; Esferas = {Flatten[{puntoInicial - {r1, 0}, r1}],
  Flatten[{puntoInicial + {r2, 0}, r2}]}];
For[i = 1, i <= 2, i++, ActCeldasBC[i]]; CapExt = {1, 2};
While[Length[CapExt] > 0, Print[Length[Esferas]]; BorrarOutput;
  piv = Last[CapExt]; CapExt = Most[CapExt]; vecinas = Vecinas[piv,
2];
  For[i = 1, i <= Length[vecinas], i++,
    r = rAl; nesf = Centro[piv, vecinas[[i]], r];
    nesf = Table[Join[nesf[[i]], {r}], {i, Length[nesf]}];
    While[Length[nesf] > 0, If[Admisible[Last[nesf], piv,
vecinas[[i]]],
      Break[]]; nesf = Delete[nesf, Length[nesf]];
    If[Length[nesf] > 0, nesf = Last[nesf]; AppendTo[Esferas, nesf];
    AppendTo[vecinas, Length[Esferas]]; AppendTo[CapExt,
      Length[Esferas]]; ActCeldasBC[Length[Esferas]]; ]; ];

```

Programa Principal 2

```

r1 = rAl; r2 = rAl; Esferas = {Flatten[{puntoInicial - {r1, 0}, r1}],
  Flatten[{puntoInicial + {r2, 0}, r2}]}];
For[i = 1, i <= 2, i++, ActCeldasBC[i]]; CapExt = {1, 2};
While[Length[CapExt] > 0, Print[Length[Esferas]]; BorrarOutput;
  piv = Last[CapExt]; vecinas = Vecinas[piv, 2]; aumento = 0;
  For[i = 1, i <= Length[vecinas], i++,
    r = rAl; nesf = Centro[piv, vecinas[[i]], r];
    nesf = Table[Join[nesf[[i]], {r}], {i, Length[nesf]}];
    While[Length[nesf] > 0, If[Admisible[Last[nesf], piv, vecinas[[i]]],
      Break[]]; nesf = Delete[nesf, Length[nesf]];
    If[Length[nesf] > 0, nesf = Last[nesf]; aumento = aumento + 1;
    AppendTo[Esferas, nesf]; AppendTo[vecinas, Length[Esferas]];
    AppendTo[CapExt, Length[Esferas]]; ActCeldasBC[Length[Esferas]];
    Break[]]; ]; If[aumento == 0, CapExt = Most[CapExt]]; ];

```

Programa para la generación de un medio con cuadrados.

Constantes

```

puntos = 10*{{0, 0}, {1, 0}, {1, 1}, {0, 1}};
segmentos = {{1, 2}, {2, 3}, {3, 4}, {4, 1}};
X = Table[puntos[[i,1]], {i, Length[puntos]}];
Y = Table[puntos[[i,2]], {i, Length[puntos]}];
margenX = 0.05*(Max[X] - Min[X]); margenY = 0.05*(Max[Y] - Min[Y]);
Xmin = Min[X] - margenX; Xmax = Max[X] + margenX; Ymin = Min[Y] - margenY;
Ymax = Max[Y] + margenY; refMallaIntX = 0.2; refMallaIntY = 0.2;
N1 = Round[1/refMallaIntX]; N2 = Round[1/refMallaIntY];
hxInt = (Xmax - Xmin)/N1; hyInt = (Ymax - Ymin)/N2;
radMin = Min[margenX, margenY]; radMax = 6*radMin;
puntoInicial = {(Xmin + Xmax)/2, (Ymin + Ymax)/2}; hxBC = 2*radMax;
hyBC = hxBC; N1BC = Round[(Xmax - Xmin)/hxBC];
N2BC = Round[(Ymax - Ymin)/hyBC]; hxBC = (Xmax - Xmin)/N1BC;
hyBC = (Ymax - Ymin)/N2BC; NxMalla = 100; NyMalla = 100;
hxmalla = (Xmax - Xmin)/NxMalla; hymalla = (Ymax - Ymin)/NyMalla;
CeldasBC = Table[{}, {i, N2BC}, {j, N1BC}];

```

Mallas

```

malla = Table[N[{Xmin + i*hxmalla, Ymin + j*hymalla}], {i, NxMalla - 1},
  {j, NyMalla - 1}];
malla = Partition[Flatten[malla], 2];
malla = Select[malla, DentroCuad[Esferas[[2]], #1] & ];
mallaGraf = Graphics[Table[Point[malla[[i]]], {i, Length[malla]}]];

```

Cálculo de la estructura de interioridad

```

PuntosSalientes = {}; For[i = 1, i <= Length[puntos],
  i++, sEnt = 0; sSal = 0; For[j = 1, j <= Length[segmentos], j++,
    If[segmentos[[j,1]] == i, sSal = j]; If[segmentos[[j,2]] == i,
      sEnt = j]; If[sEnt > 0 && sSal > 0, Break[]];
    orP = {puntos[[segmentos[[sSal,2]]]] - puntos[[segmentos[[sSal,1]]]],
      puntos[[segmentos[[sEnt,2]]]] - puntos[[segmentos[[sEnt,1]]]]};
    orP = Cross[Join[orP[[1]], {0}], Join[orP[[2]], {0}]];
    orP = orP[[3]] < 0; AppendTo[PuntosSalientes, orP]];
CeldasInt = {}; For[i = 1, i <= N2, i++, For[j = 1, j <= N1, j++,
  list = {}; Print[i, j]; For[k = 1, k <= Length[segmentos], k++,
    AppendTo[list, {k, MinCellSegm[i, j, k], MaxCellSegm[i, j, k],
      CortaCellSegm[i, j, k]}]; ]; list = Reducir[list];
  list = Table[{First[list[[k1]]], Last[list[[k1]]], {k1, Length[list]}}];
  If[Count[list, {_, False}] == Length[list],
    listAux = Table[First[list[[k1]]], {k1, Length[list]};
    list = DentroCellwrtSeg[i, j, listAux]; AppendTo[CeldasInt,
      list]; ]; CeldasInt = Partition[CeldasInt, N2];

```

Funciones

```

rA1 := Random[Real, {radMin, radMax}]; aA1 := Random[Real, {0, Pi/2}];
DPSegm[p_, i_] := Module[{s = {puntos[[segmentos[[i,1]]]],
  puntos[[segmentos[[i,2]]]]}, p1, p2, n, sol = 0},
  n = s[[2]] - s[[1]]; p1 = p - s[[1]]; p2 = p - s[[2]];
  Which[p1 . n < 0, sol = Norm[p1], p2 . n > 0, sol = Norm[p2], True,
    sol = Norm[p1 - (p1 . n/Norm[n]^2)*n]; sol = N[sol]; sol];
Reducir[l1_] := Module[{l = l1, i, j, sol},
  For[i = 1, i <= Length[l] - 1, i++, For[j = i + 1, j <= Length[l], j++,
    If[l[[i,3]] <= l[[j,2]], l = Delete[l, j]; j = j - 1; ];
    If[l[[j,3]] <= l[[i,2]], l = Delete[Insert[l, l[[j]], i], i + 1];
    l = Delete[l, j]; j = i; ]]; sol = l; sol];
HayIntSegm[seg1_, seg2_] := Module[{n1, n2, sol}, n1 = seg1[[2]] - seg1[[1]];
  n1 = {-n1[[2]], n1[[1]]}; n2 = seg2[[2]] - seg2[[1]];
  n2 = {-n2[[2]], n2[[1]]};
  sol = (seg2[[1]] - seg1[[1]]) . n1*(seg2[[2]] - seg1[[1]]) . n1 <= 0 &&
    (seg1[[1]] - seg2[[1]]) . n2*(seg1[[2]] - seg2[[1]]) . n2 <= 0 &&
    Norm[Cross[Join[seg2[[2]] - seg2[[1]], {0}],
      Join[seg1[[2]] - seg1[[1]], {0}]] > 0]; sol];
HayIntEstSegm[seg1_, seg2_] := Module[{n1, n2, sol},
  n1 = seg1[[2]] - seg1[[1]]; n1 = {-n1[[2]], n1[[1]]};
  n2 = seg2[[2]] - seg2[[1]]; n2 = {-n2[[2]], n2[[1]]};
  sol = (seg2[[1]] - seg1[[1]]) . n1*(seg2[[2]] - seg1[[1]]) . n1 < 0 &&
    (seg1[[1]] - seg2[[1]]) . n2*(seg1[[2]] - seg2[[1]]) . n2 < 0; sol];
IntSegm[seg1_, seg2_] := Module[{n1, n2, x, y, sol = {}},
  n1 = seg1[[2]] - seg1[[1]]; n1 = {-n1[[2]], n1[[1]]};
  n2 = seg2[[2]] - seg2[[1]]; n2 = {-n2[[2]], n2[[1]]};

```

```

If[HayIntEstSegm[seg1, seg2],
  sol = N[{x, y} /. FindInstance[{({x, y} - seg1[[1]]) . n1 == 0,
    ({x, y} - seg2[[1]]) . n2 == 0}, {x, y}]]]; sol];
NormCell[p_] := 2*Max[Abs[p[[1]]/hxInt], Abs[p[[2]]/hyInt]];
MinCellSegm[i_, j_, k_] := Module[{xcmin = Xmin + hxInt*(j - 1),
  xcmax = Xmin + hxInt*j, ycmin = Ymax - hyInt*i,
  ycmax = Ymax - hyInt*(i - 1), x, y, s, t, p, sol},
  Clear[x, y]; s = {puntos[[segmentos[[k,1]]]],
    puntos[[segmentos[[k,2]]]]]; p = {x, y};
  sol = NMinimize[{(s[[1]] + t*(s[[2]] - s[[1]]) - p) .
    (s[[1]] + t*(s[[2]] - s[[1]]) - p), xcmin <= x <= xcmax &&
    ycmin <= y <= ycmax && 0 <= t <= 1}, {x, y, t}];
  sol = {sol[[1]], {x, y} /. sol[[2]]]; sol];
MaxCellSegm[i_, j_, k_] := Module[{xcmin = Xmin + hxInt*(j - 1),
  xcmax = Xmin + hxInt*j, ycmin = Ymax - hyInt*i,
  ycmax = Ymax - hyInt*(i - 1), c1, c2, c3, c4, x, y, s, p, p1, p2, n,
  CondExt1, CondExt2, CondSeg, maxseg, max1, max2, sol},
  Clear[x, y]; c1 = {xcmin, ycmin}; c2 = {xcmax, ycmin};
  c3 = {xcmax, ycmax}; c4 = {xcmin, ycmax}; centro = 0.5*(c1 + c3);
  s = {puntos[[segmentos[[k,1]]]], puntos[[segmentos[[k,2]]]]];
  p = {x, y}; p1 = p - s[[1]]; p2 = p - s[[2]]; n = s[[2]] - s[[1]];
  CondExt1 = (c1 - s[[1]]) . n <= 0 || (c2 - s[[1]]) . n <= 0 ||
    (c3 - s[[1]]) . n <= 0 || (c4 - s[[1]]) . n <= 0;
  CondExt2 = (c1 - s[[2]]) . n >= 0 || (c2 - s[[2]]) . n >= 0 ||
    (c3 - s[[2]]) . n >= 0 || (c4 - s[[2]]) . n >= 0;
  CondSeg = (c1 - s[[1]]) . n >= 0 && (c1 - s[[2]]) . n <= 0 ||
    (c2 - s[[1]]) . n >= 0 && (c2 - s[[2]]) . n <= 0 ||
    (c3 - s[[1]]) . n >= 0 && (c3 - s[[2]]) . n <= 0 ||
    (c4 - s[[1]]) . n >= 0 && (c4 - s[[2]]) . n <= 0;
  If[CondSeg, maxseg = NMaximize[{(p1 - (p1 . n/Norm[n]^2)*n) .
    (p1 - (p1 . n/Norm[n]^2)*n), xcmin <= x <= xcmax &&
    ycmin <= y <= ycmax && (p - s[[1]]) . n >= 0 &&
    (p - s[[2]]) . n <= 0}, {x, y}], maxseg = {0, {x -> 0, y -> 0}}];
  maxseg = {maxseg[[1]], {x, y} /. maxseg[[2]]];
  If[CondExt1, max1 = NMaximize[{(p - s[[1]]) . (p - s[[1]]),
    xcmin <= x <= xcmax && ycmin <= y <= ycmax && (p - s[[1]]) . n <=
    0}, {x, y}], max1 = {0, {x -> 0, y -> 0}}];
  max1 = {max1[[1]], {x, y} /. max1[[2]]];
  If[CondExt2, max2 = NMaximize[{(p - s[[2]]) . (p - s[[2]]),
    xcmin <= x <= xcmax && ycmin <= y <= ycmax && (p - s[[2]]) . n >=
    0}, {x, y}], max2 = {0, {x -> 0, y -> 0}}];
  max2 = {max2[[1]], {x, y} /. max2[[2]]];
  sol = Max[maxseg[[1]], max1[[1]], max2[[1]]]; sol];
CortaCellSegm[i_, j_, k_] :=
  Module[{s = {puntos[[segmentos[[k,1]]]], puntos[[segmentos[[k,2]]]]}, t,
  centro, sol}, centro = {Xmin + hxInt*(j - 0.5), Ymax - hyInt*(i - 0.5)};
  sol = Minimize[{NormCell[s[[1]] + t*(s[[2]] - s[[1]]) - centro],
    0 <= t <= 1}, {t}][[1]]; sol = sol <= 1; sol];
DentroPointwrtSeg[p_, k_] := Module[{n, n12, p1, p2, sol},
  p1 = p - puntos[[segmentos[[k,1]]]]; p2 = p - puntos[[segmentos[[k,2]]]];
  n12 = p1 - p2; n = {-n12[[2]], n12[[1]]}; Which[p1 . n12 < 0,
    sol = PuntosSalientes[[segmentos[[k,1]]]]; sol = !sol, p2 . n12 > 0,
    sol = PuntosSalientes[[segmentos[[k,2]]]]; sol = !sol, True,
    sol = p1 . n >= 0]; sol];
DentroCellwrtSeg[i_, j_, listSeg_] :=
  Module[{c1 = {Xmin + hxInt*(j - 1), Ymax - hyInt*i}, distancias, k, sol},
  distancias = Table[DPSegm[c1, listSeg[[k]]], {k, Length[listSeg]}];

```



```

    sol = Min[distancias]; sol = First[Flatten[Position[distancias, sol]]];
    sol = listSeg[[sol]]; sol = DentroPointwrtSeg[c1, sol]; sol];
Dentro[p_] := Module[{distancias, listSeg, i, j, k, sol},
  j = Quotient[p[[1]] - Xmin, hxInt] + 1;
  i = N2 - Quotient[p[[2]] - Ymin, hyInt];
  If[1 <= i <= Length[CeldasInt] && 1 <= j <= Length[CeldasInt[[1]]],
    listSeg = CeldasInt[[i,j]]; If[Length[listSeg] > 0,
      listSeg = Table[First[listSeg[[k]]], {k, Length[listSeg]}];
      distancias = Table[DPSegm[p, listSeg[[k]]], {k, Length[listSeg]}];
      sol = Min[distancias]; sol = First[Flatten[Position[distancias,
        sol]]]; sol = listSeg[[sol]]; sol = DentroPointwrtSeg[p, sol],
      sol = listSeg, sol = False]; sol];
ActCeldasBC[i1_] := Module[{p, i, j}, p = Most[Esferas[[i1]]];
  j = Quotient[p[[1]] - Xmin, hxBC] + 1;
  i = N2BC - Quotient[p[[2]] - Ymin, hyBC];
  CeldasBC = Insert[CeldasBC, i1, {i, j, 1}]; ];
Vecinas[i1_, d_] := Module[{p, i, j, i2, j2, sol},
  p = Take[Esferas[[i1]], 2]; j = Quotient[p[[1]] - Xmin, hxBC] + 1;
  i = N2BC - Quotient[p[[2]] - Ymin, hyBC];
  sol = Table[CeldasBC[[i2,j2]], {i2, Max[1, i - d], Min[N2BC, i + d]},
    {j2, Max[1, j - d], Min[N1BC, j + d]}]; sol = Flatten[sol];
  sol = DeleteCases[sol, i1]; sol];
Admisible[esf_, ind1_, ind2_] := Module[{p, r, a, vec, i, j, i2, j2,
  cont = 1, sol}, p = Take[esf, 2]; r = esf[[3]]; a = Last[esf];
  j = Quotient[p[[1]] - Xmin, hxBC] + 1;
  i = N2BC - Quotient[p[[2]] - Ymin, hyBC]; sol = Dentro[p];
  If[sol, vec = Table[CeldasBC[[i2,j2]], {i2, Max[1, i - 1],
    Min[N2BC, i + 1]}, {j2, Max[1, j - 1], Min[N1BC, j + 1]}];
    vec = Flatten[vec]; vec = DeleteCases[vec, ind1];
    vec = DeleteCases[vec, ind2]; While[cont <= Length[vec] && sol,
      sol = sol && !Corta[esf, Esferas[[vec[[cont]]]]];
      cont = cont + 1; ]; sol]; BorrarOutput :=
Module[{nb}, nb = SelectedNotebook[]; SelectionMove[nb, Next,
  GeneratedCell]; SelectionMove[nb, All, Cell]; NotebookDelete[nb]; ];
CondNec[i_, j_, r_] := (Take[Esferas[[i]], 2] - Take[Esferas[[j]], 2]) .
  (Take[Esferas[[i]], 2] - Take[Esferas[[j]], 2]) <=
  (2*r + Esferas[[i,3]] + Esferas[[j,3]])^2;
VertCuad[cuad_] := Module[{j, sol},
  sol = Table[Take[cuad, 2] + cuad[[3]]*{Cos[cuad[[4]] + j*(Pi/2)],
    Sin[cuad[[4]] + j*(Pi/2)]}, {j, 0, 3}]; sol];
RadVect[cuad_] := Module[{i, sol},
  sol = Table[cuad[[3]]*{Cos[cuad[[4]] + i*(Pi/2)],
    Sin[cuad[[4]] + i*(Pi/2)]}, {i, 0, 3}]; sol];
SegmCuad[cuad_] := Module[{sol}, sol = VertCuad[cuad];
  sol = Table[{sol[[j]], sol[[1 + Mod[j, 4]]], {j, 4}}]; sol];
Centro[i_, j_, r_, a_] := Module[{A1 = {}, A2 = {}, rv, v, i1, j1, sol = {}},
  rv = {RadVect[Esferas[[i]], RadVect[Esferas[[j]]],
    RadVect[{0, 0, r, a}]}; v = {VertCuad[Esferas[[i]],
    VertCuad[Esferas[[j]]]}; For[i1 = 1, i1 <= 4, i1++,
  For[j1 = 1, j1 <= 4, j1++, If[rv[[1,i1]] . rv[[3,j1]] < 0,
    AppendTo[A1, {v[[1,i1]] - rv[[3,j1]], v[[1,i1]] -
      rv[[3,Mod[j1, 4] + 1]]}]; AppendTo[A1, {v[[1,i1]] - rv[[3,j1]],
      v[[1,Mod[i1, 4] + 1]] - rv[[3,j1]]}];
    If[rv[[2,i1]] . rv[[3,j1]] < 0, AppendTo[A2, {v[[2,i1]] - rv[[3,j1]],
      v[[2,i1]] - rv[[3,Mod[j1, 4] + 1]]}]; AppendTo[A2,
      {v[[2,i1]] - rv[[3,j1]], v[[2,Mod[i1, 4] + 1]] - rv[[3,j1]]}]]];
  For[i1 = 1, i1 <= Length[A1], i1++, For[j1 = 1, j1 <= Length[A2], j1++,

```

```

    sol = Join[sol, IntSegm[A1[[i1]], A2[[j1]]]]];
    sol = Select[sol, !Corta[Esferas[[i]], Join[#1, {r, a}]] & ];
    sol = Select[sol, !Corta[Esferas[[j]], Join[#1, {r, a}]] & ]; sol];
DentroCuad[cuad_, p_] := Module[{i, n, puntos = VertCuad[cuad], sol = True},
  For[i = 1, i <= 4, i++, n = puntos[[i]] - puntos[[Mod[i, 4] + 1]];
  n = {-n[[2]], n[[1]]}; sol = sol && (p - puntos[[i]]) . n < 0; sol];
MallaCuad[cuad_] := Module[{ = 0.1, p1, p2, p3, p4, rv = RadVect[cuad], i,
  N1, sol = {}}, p1 = Take[cuad, 2] + (1 - )*rv[[1]];
  p2 = Take[cuad, 2] + (1 - )*rv[[2]];
  p3 = Take[cuad, 2] + (1 - )*rv[[3]];
  p4 = Take[cuad, 2] + (1 - )*rv[[4]]; N1 = Floor[Norm[p2 - p1]/ ^2];
  For[i = 0, i <= N1, i++, AppendTo[sol, p1 + i*((p2 - p1)/N1)];
  AppendTo[sol, p2 + i*((p3 - p2)/N1)]; AppendTo[sol,
  p3 + i*((p4 - p3)/N1)]; AppendTo[sol, p4 + i*((p1 - p4)/N1)]; ];
  sol = Union[sol]; sol];
Corta[cuad1_, cuad2_] := Module[{i, mallal = MallaCuad[cuad1], sol = False},
  For[i = 1, i <= Length[mallal], i++,
  sol = sol || DentroCuad[cuad2, mallal[[i]]]; If[sol, Break[]]; ]; sol];
Graficación
PlotPuntos := ListPlot[puntos, PlotStyle -> PointSize[0.02],
  DisplayFunction -> Identity];
PlotSegm := Graphics[Table[{Thickness[0.006], Hue[1],
  Line[{puntos[[segmentos[[i,1]]]], puntos[[segmentos[[i,2]]]]}],
  {i, Length[segmentos]}]];
PlotEtiquetas := Module[{i, sol}, sol = {}; For[i = 1, i <= Length[puntos],
  i++, AppendTo[sol, Text[ToString[i], puntos[[i]], {0, -1}]]];
  For[i = 1, i <= Length[segmentos], i++, AppendTo[sol,
  Text[ToString[i], (puntos[[segmentos[[i,1]]]] +
  puntos[[segmentos[[i,2]]]])/2, {0, -1}]]];
  For[i = 1, i <= Length[Esferas], i++, AppendTo[sol,
  Text[ToString[i], Take[Esferas[[i]], 2]]]; sol = Graphics[sol]; sol];
PlotCeldasInt := Graphics[
  Join[Table[Line[{{Xmin + i*hxInt, Ymin}, {Xmin + i*hxInt, Ymax}}],
  {i, 0, N1}], Table[Line[{{Xmin, Ymin + i*hyInt},
  {Xmax, Ymin + i*hyInt}}], {i, 0, N2}]]];
PlotCeldasBC := Graphics[
  Join[Table[Line[{{Xmin + i*hxBC, Ymin}, {Xmin + i*hxBC, Ymax}}],
  {i, 0, N1BC}], Table[Line[{{Xmin, Ymin + i*hyBC},
  {Xmax, Ymin + i*hyBC}}], {i, 0, N2BC}]]];
PlotEsferas := Module[{i, j, sol},
  sol = Table[Line[Table[Take[Esferas[[i]], 2] + Esferas[[i,3]]*
  {Cos[Esferas[[i,4]] + (Pi/2)*Mod[j, 4]],
  Sin[Esferas[[i,4]] + (Pi/2)*Mod[j, 4]], {j, 0, 4}}],
  {i, Length[Esferas]}]; sol = Graphics[sol]; sol];
GrafCentrosBC[i_, j_] := Module[{k, sol = {}},
  For[k = 1, k <= Length[Esferas], k++, If[Count[CeldasBC[[i,j]], k] > 0,
  AppendTo[sol, k]]; sol = Table[{Hue[1], PointSize[0.02],
  Point[Most[Esferas[[sol[[k]]]]]], {k, Length[sol]}}];
  sol = Graphics[sol]; sol];
GrafTodo := Show[PlotEsferas, PlotSegm, AspectRatio -> Automatic,
  DisplayFunction -> $DisplayFunction, Axes -> False, PlotRange -> All];
Programa Principal
r1 = rAl; a1 = 0; r2 = rAl; a2 = 0;
Esferas = {Flatten[{puntoInicial - {r1*Cos[a1], 0}, r1, a1}],
  Flatten[{puntoInicial + {r2*Cos[a2], 0}, r2, a2}]}];
For[i = 1, i <= 2, i++, ActCeldasBC[i]; CapExt = {1, 2};
  While[Length[CapExt] > 0, Print[Length[Esferas]]; Null; piv = Last[CapExt];

```

```

vecinas = Vecinas[piv, 2]; aumento = 0; For[i = 1, i <= Length[vecinas],
i++, r = rA1; a = aA1; nesf = Centro[piv, vecinas[[i]], r, a];
nesf = Table[Join[nesf[[i1]], {r, a}], {i1, Length[nesf]}];
While[Length[nesf] > 0, If[Admisible[Last[nesf], piv, vecinas[[i]]],
Break[]]; nesf = Most[nesf]; If[Length[nesf] > 0,
nesf = Last[nesf]; aumento = aumento + 1; AppendTo[Esferas, nesf];
AppendTo[vecinas, Length[Esferas]]; AppendTo[CapExt,
Length[Esferas]]; ActCeldasBC[Length[Esferas]]; Break[]]; ];
If[aumento == 0, CapExt = Most[CapExt]]; ];

```

Programa para la generación de un medio con cuadrados.

```

<< "Geometry`Polytopes`"; << "Geometry`Rotations`";
Constantes
puntos = 50*{{0, 0}, {1, 0}, {1, 1}, {0, 1}}; segmentos = {{1, 2}, {2, 3},
{3, 4}, {4, 1}};
X = Table[puntos[[i,1]], {i, Length[puntos]}]; Y = Table[puntos[[i,2]], {i,
Length[puntos]}];
margenX = 0.05*(Max[X] - Min[X]); margenY = 0.05*(Max[Y] - Min[Y]);
Xmin = Min[X] - margenX; Xmax = Max[X] + margenX; Ymin = Min[Y] - margenY;
Ymax = Max[Y] + margenY; refMallaIntX = 0.2;
refMallaIntY = 0.2; N1 = Round[1/refMallaIntX]; N2 = Round[1/refMallaIntY];
hxInt = (Xmax - Xmin)/N1;
hyInt = (Ymax - Ymin)/N2; radMin = 2; radMax = 6*radMin; puntoInicial =
{(Xmin + Xmax)/2, (Ymin + Ymax)/2}; hxBc = 2*radMax;
hyBc = hxBc; N1Bc = Round[(Xmax - Xmin)/hxBc]; N2Bc = Round[(Ymax -
Ymin)/hyBc]; hxBc = (Xmax - Xmin)/N1Bc;
hyBc = (Ymax - Ymin)/N2Bc; NxMalla = 100; NyMalla = 100; hxmalla = (Xmax -
Xmin)/NxMalla; hymalla = (Ymax - Ymin)/NyMalla;
CeldasBC = Table[{}, {i, N2Bc}, {j, N1Bc}]; vertT = N[Vertices[Triangle]];
vertC = N[Vertices[Square]];
vertP = N[Vertices[Pentagon]]; pol0 = {vertT, vertC, vertP};
Mallas
malla = Table[N[{Xmin + i*hxmalla, Ymin + j*hymalla}], {i, NxMalla - 1}, {j,
NyMalla - 1}];
malla = Partition[Flatten[malla], 2];
malla = Select[malla, DentroCuad[Esferas[[2]], #1] & ]; mallaGraf =
Graphics[Table[Point[malla[[i]]], {i, Length[malla]}]];
Cálculo de la estructura de interioridad
t0 = SessionTime[]; PuntosSalientes = {}; For[i = 1, i <= Length[puntos], i++,
sEnt = 0; sSal = 0; For[j = 1, j <= Length[segmentos], j++,
If[segmentos[[j,1]] == i, sSal = j];
If[segmentos[[j,2]] == i, sEnt = j]; If[sEnt > 0 && sSal > 0, Break[]];
orP = {puntos[[segmentos[[sSal,2]]]] - puntos[[segmentos[[sSal,1]]]],
puntos[[segmentos[[sEnt,2]]]] -
puntos[[segmentos[[sEnt,1]]]]}; orP = Cross[Join[orP[[1]], {0}],
Join[orP[[2]], {0}]]; orP = orP[[3]] < 0;
AppendTo[PuntosSalientes, orP];
CeldasInt = {}; For[i = 1, i <= N2, i++, For[j = 1, j <= N1, j++, list = {};
Print[i, j];
For[k = 1, k <= Length[segmentos], k++, AppendTo[list, {k, MinCellSegm[i,
j, k], MaxCellSegm[i, j, k],
CortaCellSegm[i, j, k]}]; ]; list = Reducir[list]; list =
Table[{First[list[[k1]]], Last[list[[k1]]],
{k1, Length[list]}}; If[Count[list, {_, False}] == Length[list],

```

```

listAux = Table[First[list[[k1]]], {k1, Length[list]}; list =
DentroCellwrtSeg[i, j, listAux];
AppendTo[CeldasInt, list]; ]; CeldasInt = Partition[CeldasInt, N2];
Funciones
rA1 := Random[Real, {radMin, radMax}]; aA1 := Random[Real, {0, Pi/2}];
DPSegm[p_, i_] := Module[{s = {puntos[[segmentos[[i,1]]]],
puntos[[segmentos[[i,2]]]]}, p1, p2, n, sol = 0},
n = s[[2]] - s[[1]]; p1 = p - s[[1]]; p2 = p - s[[2]]; Which[p1 . n < 0,
sol = Norm[p1], p2 . n > 0, sol = Norm[p2], True,
sol = Norm[p1 - (p1 . n/Norm[n]^2)*n]]; sol = N[sol]; sol];
Reducir[l1_] := Module[{l = l1, i, j, sol}, For[i = 1, i <= Length[l] - 1,
i++, For[j = i + 1, j <= Length[l], j++,
If[l[[i,3]] <= l[[j,2]], l = Delete[l, j]; j = j - 1; ]; If[l[[j,3]] <=
l[[i,2]],
l = Delete[Insert[l, l[[j]], i], i + 1]; l = Delete[l, j]; j = i;
]]; sol = l; sol];
HayIntSegm[seg1_, seg2_] := Module[{n1, n2, sol}, n1 = seg1[[2]] - seg1[[1]];
n1 = {-n1[[2]], n1[[1]]};
n2 = seg2[[2]] - seg2[[1]]; n2 = {-n2[[2]], n2[[1]]};
sol = (seg2[[1]] - seg1[[1]]) . n1*(seg2[[2]] - seg1[[1]]) . n1 <= 0 &&
(seg1[[1]] - seg2[[1]]) . n2*(seg1[[2]] - seg2[[1]]) . n2 <= 0 &&
Norm[Cross[Join[seg2[[2]] - seg2[[1]], {0}], Join[seg1[[2]] -
seg1[[1]], {0}]] > 0]; sol];
HayIntEstSegm[seg1_, seg2_] := Module[{n1, n2, sol}, n1 = seg1[[2]] -
seg1[[1]]; n1 = {-n1[[2]], n1[[1]]};
n2 = seg2[[2]] - seg2[[1]]; n2 = {-n2[[2]], n2[[1]]};
sol = (seg2[[1]] - seg1[[1]]) . n1*(seg2[[2]] - seg1[[1]]) . n1 < 0 &&
(seg1[[1]] - seg2[[1]]) . n2*(seg1[[2]] - seg2[[1]]) . n2 < 0; sol];
IntSegm[seg1_, seg2_] := Module[{n1, n2, x, y, sol = {}}, n1 = seg1[[2]] -
seg1[[1]]; n1 = {-n1[[2]], n1[[1]]};
n2 = seg2[[2]] - seg2[[1]]; n2 = {-n2[[2]], n2[[1]]};
If[HayIntEstSegm[seg1, seg2],
sol = N[{x, y} /. FindInstance[{{(x, y) - seg1[[1]]) . n1 == 0, ((x, y)
- seg2[[1]]) . n2 == 0}, {x, y}]]]; sol];
NormCell[p_] := 2*Max[Abs[p[[1]]/hxInt], Abs[p[[2]]/hyInt]]; MinCellSegm[i_,
j_, k_] :=
Module[{xcmin = Xmin + hxInt*(j - 1), xcmax = Xmin + hxInt*j, ycmin = Ymax
- hyInt*i, ycmax = Ymax - hyInt*(i - 1), x, y,
s, t, p, sol}, Clear[x, y]; s = {puntos[[segmentos[[k,1]]]],
puntos[[segmentos[[k,2]]]]}; p = {x, y};
sol = NMinimize[{(s[[1]] + t*(s[[2]] - s[[1])) - p) . (s[[1]] + t*(s[[2]]
- s[[1])) - p),
xcmin <= x <= xcmax && ycmin <= y <= ycmax && 0 <= t <= 1}, {x, y,
t}]; sol = {sol[[1]], {x, y} /. sol[[2]]}; sol];
MaxCellSegm[i_, j_, k_] := Module[{xcmin = Xmin + hxInt*(j - 1), xcmax =
Xmin + hxInt*j, ycmin = Ymax - hyInt*i,
ycmax = Ymax - hyInt*(i - 1), c1, c2, c3, c4, x, y, s, p, p1, p2, n,
CondExt1, CondExt2, CondSeg, maxseg, maxl, max2,
sol}, Clear[x, y]; c1 = {xcmin, ycmin}; c2 = {xcmax, ycmin}; c3 = {xcmax,
ycmax}; c4 = {xcmin, ycmax};
centro = 0.5*(c1 + c3); s = {puntos[[segmentos[[k,1]]]],
puntos[[segmentos[[k,2]]]]}; p = {x, y}; p1 = p - s[[1]];
p2 = p - s[[2]]; n = s[[2]] - s[[1]]; CondExt1 = (c1 - s[[1]]) . n <= 0
|| (c2 - s[[1]]) . n <= 0 ||
(c3 - s[[1]]) . n <= 0 || (c4 - s[[1]]) . n <= 0; CondExt2 = (c1 -
s[[2]]) . n >= 0 || (c2 - s[[2]]) . n >= 0 ||

```

```

(c3 - s[[2]]) . n >= 0 || (c4 - s[[2]]) . n >= 0; CondSeg = (c1 -
s[[1]]) . n >= 0 && (c1 - s[[2]]) . n <= 0 ||
(c2 - s[[1]]) . n >= 0 && (c2 - s[[2]]) . n <= 0 || (c3 - s[[1]]) . n
>= 0 && (c3 - s[[2]]) . n <= 0 ||
(c4 - s[[1]]) . n >= 0 && (c4 - s[[2]]) . n <= 0;
If[CondSeg, maxseg = NMaximize[{(p1 - (p1 . n/Norm[n]^2)*n) . (p1 - (p1 .
n/Norm[n]^2)*n),
xcmin <= x <= xmax && ycmin <= y <= ymax && (p - s[[1]]) . n >= 0
&& (p - s[[2]]) . n <= 0}, {x, y}],
maxseg = {0, {x -> 0, y -> 0}}]; maxseg = {maxseg[[1]], {x, y} /.
maxseg[[2]]};
If[CondExt1, max1 = NMaximize[{(p - s[[1]]) . (p - s[[1]]), xcmin <= x <=
xmax && ycmin <= y <= ymax &&
(p - s[[1]]) . n <= 0}, {x, y}], max1 = {0, {x -> 0, y -> 0}}]; max1
= {max1[[1]], {x, y} /. max1[[2]]};
If[CondExt2, max2 = NMaximize[{(p - s[[2]]) . (p - s[[2]]), xcmin <= x <=
xmax && ycmin <= y <= ymax &&
(p - s[[2]]) . n >= 0}, {x, y}], max2 = {0, {x -> 0, y -> 0}}]; max2
= {max2[[1]], {x, y} /. max2[[2]]};
sol = Max[maxseg[[1]], max1[[1]], max2[[1]]]; sol];
CortaCellSegm[i_, j_, k_] := Module[{s = {puntos[[segmentos[[k,1]]]],
puntos[[segmentos[[k,2]]]]}, t, centro, sol},
centro = {Xmin + hxInt*(j - 0.5), Ymax - hyInt*(i - 0.5)};
sol = Minimize[{NormCell[s[[1]] + t*(s[[2]] - s[[1]]) - centro], 0 <= t
<= 1}, {t}][[1]]; sol = sol <= 1; sol];
DentroPointwrtSeg[p_, k_] := Module[{n, n12, p1, p2, sol}, p1 = p -
puntos[[segmentos[[k,1]]]];
p2 = p - puntos[[segmentos[[k,2]]]]; n12 = p1 - p2; n = {-n12[[2]],
n12[[1]]};
Which[p1 . n12 < 0, sol = PuntosSalientes[[segmentos[[k,1]]]]; sol =
!sol, p2 . n12 > 0,
sol = PuntosSalientes[[segmentos[[k,2]]]]; sol = !sol, True, sol = p1 .
n >= 0]; sol];
DentroCellwrtSeg[i_, j_, listSeg_] := Module[{c1 = {Xmin + hxInt*(j - 1), Ymax
- hyInt*i}, distancias, k, sol},
distancias = Table[DPSegm[c1, listSeg[[k]]], {k, Length[listSeg]}]; sol =
Min[distancias];
sol = First[Flatten[Position[distancias, sol]]]; sol = listSeg[[sol]];
sol = DentroPointwrtSeg[c1, sol]; sol];
Dentro[p_] := Module[{distancias, listSeg, i, j, k, sol}, j = Quotient[p[[1]]
- Xmin, hxInt] + 1;
i = N2 - Quotient[p[[2]] - Ymin, hyInt]; If[1 <= i <= Length[CeldasInt]
&& 1 <= j <= Length[CeldasInt][[1]],
listSeg = CeldasInt[[i,j]]; If[Length[listSeg] > 0, listSeg =
Table[First[listSeg[[k]]], {k, Length[listSeg]}];
distancias = Table[DPSegm[p, listSeg[[k]]], {k, Length[listSeg]}];
sol = Min[distancias];
sol = First[Flatten[Position[distancias, sol]]]; sol =
listSeg[[sol]]; sol = DentroPointwrtSeg[p, sol],
sol = listSeg, sol = False]; sol];
CentroMasa[pol_] := Mean[pol];
ActCeldasBC[i1_] := Module[{p, i, j}, p = CentroMasa[Esferas[[i1]]]; j =
Quotient[p[[1]] - Xmin, hxBC] + 1;
i = N2BC - Quotient[p[[2]] - Ymin, hyBC]; CeldasBC = Insert[CeldasBC, i1,
{i, j, 1}]; ];
Vecinas[i1_, d_] := Module[{p, i, j, i2, j2, sol}, p =
CentroMasa[Esferas[[i1]]]; j = Quotient[p[[1]] - Xmin, hxBC] + 1;

```

```

i = N2BC - Quotient[p[[2]] - Ymin, hyBC]; sol = Table[CeldasBC[[i2,j2]],
{i2, Max[1, i - d], Min[N2BC, i + d]},
{j2, Max[1, j - d], Min[N1BC, j + d]}; sol = Flatten[sol]; sol =
DeleteCases[sol, {}]; sol];
Admisible[esf_, ind1_, ind2_] := Module[{p, vec, i, j, i2, j2, cont = 1, sol},
p = CentroMasa[esf]; j = Quotient[p[[1]] - Xmin, hxBC] + 1; i = N2BC -
Quotient[p[[2]] - Ymin, hyBC]; sol = Dentro[p];
If[sol, vec = Table[CeldasBC[[i2,j2]], {i2, Max[1, i - 1], Min[N2BC, i +
1]}, {j2, Max[1, j - 1], Min[N1BC, j + 1]}];
vec = Flatten[vec]; vec = DeleteCases[vec, {}]; ind1; vec =
DeleteCases[vec, ind2]; While[cont <= Length[vec] && sol,
sol = sol && !Corta[esf, Esferas[[vec[[cont]]]]]; cont = cont + 1;
]; sol];
BorrarOutput := Module[{nb}, nb = SelectedNotebook[]; SelectionMove[nb, Next,
GeneratedCell]; SelectionMove[nb, All, Cell];
NotebookDelete[nb]; ]; CondNec[i_, j_, r_] := (CentroMasa[Esferas[[i]]] -
CentroMasa[Esferas[[j]]]) .
(CentroMasa[Esferas[[i]]] - CentroMasa[Esferas[[j]]]) <=
(2*r + Norm[RadVect[Esferas[[i]]] + Norm[RadVect[Esferas[[j]]]])^2;
RadVect[pol_] := Module[{c = CentroMasa[pol], i, sol = pol}, sol =
Table[sol[[i]] - c, {i, Length[sol]}]; sol];
SegmCuad[cuad_] := Module[{j}, Table[{cuad[[j]], cuad[[1 + Mod[j,
Length[cuad]]]}], {j, Length[cuad]}];
DentroCuad[cuad_, p_] := Module[{i, n, cantVert = Length[cuad], puntos = cuad,
sol = True},
For[i = 1, i <= cantVert, i++, n = puntos[[i]] - puntos[[Mod[i, cantVert]
+ 1]]; n = {-n[[2]], n[[1]]};
sol = sol && (p - puntos[[i]]) . n < 0; sol];
MallaCuad[cuad1_] := Module[{cuad = cuad1, c = CentroMasa[cuad1], = 0.1, i,
j, p1, p2, N1, sol = {}},
N1 = Floor[Norm[cuad[[1]] - cuad[[2]]/ ]; cuad = Table[c + (1 -
)*(cuad[[i]] - c), {i, Length[cuad]}];
For[i = 1, i <= Length[cuad], i++, p1 = cuad[[i]]; p2 = cuad[[Mod[i,
Length[cuad]] + 1]];
For[j = 0, j <= N1, j++, AppendTo[sol, p1 + (j/N1)*(p2 - p1)]]; sol =
Union[sol]; sol];
GenerarPol := Module[{i, i1, k, , sol}, i = Random[Integer, {1, 3}]; k = rA1;
= aA1; sol = pol0[[i]];
sol = Table[k*Rotate2D[sol[[i1]], ], {i1, Length[sol]}]; sol];
Corta[cuad1_, cuad2_] := Module[{i, malla1 = MallaCuad[cuad1], sol = False},
For[i = 1, i <= Length[malla1], i++, sol = sol || DentroCuad[cuad2,
malla1[[i]]]; If[sol, Break[]]; ]; sol];
Centro[i_, j_, pol_] := Module[{A1 = {}, A2 = {}, rv = {RadVect[Esferas[[i]],
RadVect[Esferas[[j]]], RadVect[pol]}},
v = {Esferas[[i]], Esferas[[j]], i1, j1, newPol = {}, sol = {}},
For[i1 = 1, i1 <= Length[Esferas[[i]], i1++, For[j1 = 1, j1 <=
Length[pol], j1++,
If[rv[[1,i1]] . rv[[3,j1]] < 0, AppendTo[A1, {v[[1,i1]] - rv[[3,j1]],
v[[1,i1]] - rv[[3,Mod[j1, Length[pol]] + 1]]];
AppendTo[A1, {v[[1,i1]] - rv[[3,j1]], v[[1,Mod[i1,
Length[Esferas[[i]]] + 1]] - rv[[3,j1]]}]; ];
For[i1 = 1, i1 <= Length[Esferas[[j]], i1++, For[j1 = 1, j1 <=
Length[pol], j1++,
If[rv[[2,i1]] . rv[[3,j1]] < 0, AppendTo[A2, {v[[2,i1]] - rv[[3,j1]],
v[[2,i1]] - rv[[3,Mod[j1, Length[pol]] + 1]]];
AppendTo[A2, {v[[2,i1]] - rv[[3,j1]], v[[2,Mod[i1,
Length[Esferas[[j]]] + 1]] - rv[[3,j1]]}]]];

```

```

    For[i1 = 1, i1 <= Length[A1], i1++, For[j1 = 1, j1 <= Length[A2], j1++,
sol = Join[sol, IntSegm[A1[[i1]], A2[[j1]]]]];
    For[i1 = 1, i1 <= Length[sol], i1++, AppendTo[newPol, Table[pol[[j1]] +
sol[[i1]], {j1, Length[pol]}]]; sol = newPol;
    sol = Select[sol, !Corta[Esferas[[i]], #1] & ]; sol = Select[sol,
!Corta[Esferas[[j]], #1] & ]; sol];
Corta[cuad1_, cuad2_] := Module[{i, mallal = MallaCuad[cuad1], sol = False},
    For[i = 1, i <= Length[mallal], i++, sol = sol || DentroCuad[cuad2,
mallal[[i]]]; If[sol, Break[]]; ]; sol];

```

Graficación

```

PlotPuntos := ListPlot[puntos, PlotStyle -> PointSize[0.02], DisplayFunction
-> Identity];
PlotSegm := Graphics[Table[{Thickness[0.006], Hue[1],
Line[{puntos[[segmentos[[i,1]]], puntos[[segmentos[[i,2]]]}],
{i, Length[segmentos]}]],
PlotEtiquetas := Module[{i, sol}, sol = {}; For[i = 1, i <= Length[puntos],
i++,
    AppendTo[sol, Text[ToString[i], puntos[[i]], {0, -1}]]; For[i = 1, i <=
Length[segmentos], i++,
    AppendTo[sol, Text[ToString[i], (puntos[[segmentos[[i,1]]] +
puntos[[segmentos[[i,2]]])/2, {0, -1}]];
    For[i = 1, i <= Length[Esferas], i++, AppendTo[sol, Text[ToString[i],
CentroMasa[Esferas[[i]]]]]; sol = Graphics[sol];
    sol]; PlotCeldasInt := Graphics[Join[Table[Line[{{Xmin + i*hxInt, Ymin},
{Xmin + i*hxInt, Ymax}}], {i, 0, N1}],
    Table[Line[{{Xmin, Ymin + i*hyInt}, {Xmax, Ymin + i*hyInt}}], {i, 0,
N2}]]];
PlotCeldasBC := Graphics[Join[Table[Line[{{Xmin + i*hxBc, Ymin}, {Xmin +
i*hxBc, Ymax}}], {i, 0, N1BC}],
    Table[Line[{{Xmin, Ymin + i*hyBC}, {Xmax, Ymin + i*hyBC}}], {i, 0,
N2BC}]]];
GrafCentrosBC[i_, j_] := Module[{k, sol = {}}, For[k = 1, k <=
Length[Esferas], k++, If[Count[CeldasBC[[i,j]], k] > 0,
    AppendTo[sol, k]]; sol = Table[{Hue[1], PointSize[0.02],
Point[Most[Esferas[[sol[[k]]]]], {k, Length[sol]}];
    sol = Graphics[sol]; sol]; PlotEsferas := Module[{i, j, sol},
    sol = Table[Line[Join[Esferas[[i]], {Esferas[[i,1]]}], {i,
Length[Esferas]}]; sol = Graphics[sol]; sol];
GrafTodo := Show[PlotEsferas, PlotSegm, PlotEtiquetas, AspectRatio ->
Automatic, DisplayFunction -> $DisplayFunction,
    Axes -> False, PlotRange -> All];

```

Programa Principal

```

Esferas = {{{25., 27.30020970609815}, {22.69979029390185, 27.30020970609815},
{20.3995805878037, 25.},
    {25, 22.69979029390185}}, {{28.67430377312043, 25.}, {26.837151886560214,
26.837151886560214}, {25., 25.},
    {26.837151886560214, 23.162848113439782}}];
For[i = 1, i <= 2, i++, ActCeldasBC[i]; CapExt = {1, 2};
While[Length[CapExt] > 0, Print[Length[Esferas]]; piv = Last[CapExt]; vecinas
= Vecinas[piv, 2]; aumento = 0;
    For[i = 1, i <= Length[vecinas], i++, pol = GenerarPol; nesf = Centro[piv,
vecinas[[i]], pol];
    While[Length[nesf] > 0, If[Admisible[Last[nesf], piv, vecinas[[i]]],
Break[]]; nesf = Most[nesf];
    If[Length[nesf] > 0, nesf = Last[nesf]; aumento = aumento + 1;
BorrarOutput; AppendTo[Esferas, nesf];

```

```

AppendTo[vecinas, Length[Esferas]]; AppendTo[CapExt, Length[Esferas]];
ActCeldasBC[Length[Esferas]]; Break[]]; ];
If[aumento == 0, CapExt = Most[CapExt]]; ];

```

Programa para la generación de un medio con partículas formadas por colisión de dos discos.

Constantes

```

puntos = 10*{{0, 0}, {1, 0}, {1, 1}, {0, 1}};
segmentos = {{1, 2}, {2, 3}, {3, 4}, {4, 1}};
X = Table[puntos[[i,1]], {i, Length[puntos]}];
Y = Table[puntos[[i,2]], {i, Length[puntos]}];
margenX = 0.05*(Max[X] - Min[X]); margenY = 0.05*(Max[Y] - Min[Y]);
Xmin = Min[X] - margenX; Xmax = Max[X] + margenX; Ymin = Min[Y] - margenY;
Ymax = Max[Y] + margenY; refMallaIntX = 0.2; refMallaIntY = 0.2;
N1 = Round[1/refMallaIntX]; N2 = Round[1/refMallaIntY];
hxInt = (Xmax - Xmin)/N1; hyInt = (Ymax - Ymin)/N2;
radMin = 0.2*Min[margenX, margenY]; radMax = 6*radMin;
puntoInicial = {(Xmin + Xmax)/2, (Ymin + Ymax)/2}; hxBC = 4*radMax;
hyBC = hxBC; N1BC = Round[(Xmax - Xmin)/hxBC];
N2BC = Round[(Ymax - Ymin)/hyBC]; hxBC = (Xmax - Xmin)/N1BC;
hyBC = (Ymax - Ymin)/N2BC; NxMalla = 100; NyMalla = 100;
hxmalla = (Xmax - Xmin)/NxMalla; hymalla = (Ymax - Ymin)/NyMalla;
CeldasBC = Table[{}, {i, N2BC}, {j, N1BC}];

```

Cálculo de la estructura de interioridad

```

Null
t0 = SessionTime[]; PuntosSalientes = {}; For[i = 1, i <= Length[puntos],
i++, sEnt = 0; sSal = 0; For[j = 1, j <= Length[segmentos], j++,
If[segmentos[[j,1]] == i, sSal = j]; If[segmentos[[j,2]] == i,
sEnt = j]; If[sEnt > 0 && sSal > 0, Break[]]];
orP = {puntos[[segmentos[[sSal,2]]]] - puntos[[segmentos[[sSal,1]]]],
puntos[[segmentos[[sEnt,2]]]] - puntos[[segmentos[[sEnt,1]]]]};
orP = Cross[Join[orP[[1]], {0}], Join[orP[[2]], {0}]];
orP = orP[[3]] < 0; AppendTo[PuntosSalientes, orP]];
CeldasInt = {}; For[i = 1, i <= N2, i++, For[j = 1, j <= N1, j++,
list = {}; Print[i, j]; For[k = 1, k <= Length[segmentos], k++,
AppendTo[list, {k, MinCellSegm[i, j, k], MaxCellSegm[i, j, k],
CortaCellSegm[i, j, k]}]; ]; list = Reducir[list];
list = Table[{First[list[[k1]]], Last[list[[k1]]]}, {k1, Length[list]}];
If[Count[list, {_, False}] == Length[list],
listAux = Table[First[list[[k1]]], {k1, Length[list]}];
list = DentroCellwrtSeg[i, j, listAux]; AppendTo[CeldasInt,
list]; ]; CeldasInt = Partition[CeldasInt, N2];

```

Funciones

```

rA1 := Random[Real, {radMin, radMax}]; aA1 := Random[Real, {0, Pi/2}];
DPSegm[p_, i_] := Module[{s = {puntos[[segmentos[[i,1]]]],
puntos[[segmentos[[i,2]]]]}, p1, p2, n, sol = 0},
n = s[[2]] - s[[1]]; p1 = p - s[[1]]; p2 = p - s[[2]];
Which[p1 . n < 0, sol = Norm[p1], p2 . n > 0, sol = Norm[p2], True,
sol = Norm[p1 - (p1 . n/Norm[n]^2)*n]]; sol = N[sol]; sol];
Reducir[l1_] := Module[{l = l1, i, j, sol},
For[i = 1, i <= Length[l] - 1, i++, For[j = i + 1, j <= Length[l], j++,
If[l[[i,3]] <= l[[j,2]], l = Delete[l, j]; j = j - 1; ];

```



```

        If[l[[j,3]] <= l[[i,2]], l = Delete[Insert[l, l[[j]], i], i + 1];
        l = Delete[l, j]; j = i; ]]; sol = l; sol];
HayIntSegm[seg1_, seg2_] := Module[{n1, n2, sol}, n1 = seg1[[2]] - seg1[[1]];
n1 = {-n1[[2]], n1[[1]]}; n2 = seg2[[2]] - seg2[[1]];
n2 = {-n2[[2]], n2[[1]]};
sol = (seg2[[1]] - seg1[[1]]) . n1*(seg2[[2]] - seg1[[1]]) . n1 <= 0 &&
(seg1[[1]] - seg2[[1]]) . n2*(seg1[[2]] - seg2[[1]]) . n2 <= 0 &&
Norm[Cross[Join[seg2[[2]] - seg2[[1]], {0}],
Join[seg1[[2]] - seg1[[1]], {0}]] > 0]; sol];
HayIntEstSegm[seg1_, seg2_] := Module[{n1, n2, sol},
n1 = seg1[[2]] - seg1[[1]]; n1 = {-n1[[2]], n1[[1]]};
n2 = seg2[[2]] - seg2[[1]]; n2 = {-n2[[2]], n2[[1]]};
sol = (seg2[[1]] - seg1[[1]]) . n1*(seg2[[2]] - seg1[[1]]) . n1 < 0 &&
(seg1[[1]] - seg2[[1]]) . n2*(seg1[[2]] - seg2[[1]]) . n2 < 0; sol];
IntSegm[seg1_, seg2_] := Module[{n1, n2, x, y, sol = {}},
n1 = seg1[[2]] - seg1[[1]]; n1 = {-n1[[2]], n1[[1]]};
n2 = seg2[[2]] - seg2[[1]]; n2 = {-n2[[2]], n2[[1]]};
If[HayIntEstSegm[seg1, seg2],
sol = N[{x, y} /. FindInstance[{({x, y} - seg1[[1])) . n1 == 0,
({x, y} - seg2[[1])) . n2 == 0}, {x, y}]]]; sol];
NormCell[p_] := 2*Max[Abs[p[[1]]/hxInt], Abs[p[[2]]/hyInt]];
MinCellSegm[i_, j_, k_] := Module[{xcmin = Xmin + hxInt*(j - 1),
xcmax = Xmin + hxInt*j, ycmin = Ymax - hyInt*i,
ycmax = Ymax - hyInt*(i - 1), x, y, s, t, p, sol},
Clear[x, y]; s = {puntos[[segmentos[[k,1]]]],
puntos[[segmentos[[k,2]]]]]; p = {x, y};
sol = NMinimize[{(s[[1]] + t*(s[[2]] - s[[1])) - p) .
(s[[1]] + t*(s[[2]] - s[[1])) - p), xcmin <= x <= xcmax &&
ycmin <= y <= ycmax && 0 <= t <= 1}, {x, y, t}];
sol = {sol[[1]], {x, y} /. sol[[2]]}; sol];
MaxCellSegm[i_, j_, k_] := Module[{xcmin = Xmin + hxInt*(j - 1),
xcmax = Xmin + hxInt*j, ycmin = Ymax - hyInt*i,
ycmax = Ymax - hyInt*(i - 1), c1, c2, c3, c4, x, y, s, p, p1, p2, n,
CondExt1, CondExt2, CondSeg, maxseg, max1, max2, sol},
Clear[x, y]; c1 = {xcmin, ycmin}; c2 = {xcmax, ycmin};
c3 = {xcmax, ycmax}; c4 = {xcmin, ycmax}; centro = 0.5*(c1 + c3);
s = {puntos[[segmentos[[k,1]]]], puntos[[segmentos[[k,2]]]]];
p = {x, y}; p1 = p - s[[1]]; p2 = p - s[[2]]; n = s[[2]] - s[[1]];
CondExt1 = (c1 - s[[1]]) . n <= 0 || (c2 - s[[1]]) . n <= 0 ||
(c3 - s[[1]]) . n <= 0 || (c4 - s[[1]]) . n <= 0;
CondExt2 = (c1 - s[[2]]) . n >= 0 || (c2 - s[[2]]) . n >= 0 ||
(c3 - s[[2]]) . n >= 0 || (c4 - s[[2]]) . n >= 0;
CondSeg = (c1 - s[[1]]) . n >= 0 && (c1 - s[[2]]) . n <= 0 ||
(c2 - s[[1]]) . n >= 0 && (c2 - s[[2]]) . n <= 0 ||
(c3 - s[[1]]) . n >= 0 && (c3 - s[[2]]) . n <= 0 ||
(c4 - s[[1]]) . n >= 0 && (c4 - s[[2]]) . n <= 0;
If[CondSeg, maxseg = NMaximize[{(p1 - (p1 . n/Norm[n]^2)*n) .
(p1 - (p1 . n/Norm[n]^2)*n), xcmin <= x <= xcmax &&
ycmin <= y <= ycmax && (p - s[[1]]) . n >= 0 &&
(p - s[[2]]) . n <= 0}, {x, y}], maxseg = {0, {x -> 0, y -> 0}}];
maxseg = {maxseg[[1]], {x, y} /. maxseg[[2]]};
If[CondExt1, max1 = NMaximize[{(p - s[[1]]) . (p - s[[1]]),
xcmin <= x <= xcmax && ycmin <= y <= ycmax && (p - s[[1]]) . n <=
0}, {x, y}], max1 = {0, {x -> 0, y -> 0}}];
max1 = {max1[[1]], {x, y} /. max1[[2]]};
If[CondExt2, max2 = NMaximize[{(p - s[[2]]) . (p - s[[2]]),
xcmin <= x <= xcmax && ycmin <= y <= ycmax && (p - s[[2]]) . n >=

```

```

0}, {x, y}], max2 = {0, {x -> 0, y -> 0}}];
max2 = {max2[[1]], {x, y} /. max2[[2]]};
sol = Max[maxseg[[1]], max1[[1]], max2[[1]]]; sol];
CortaCellSegm[i_, j_, k_] :=
Module[{s = {puntos[[segmentos[[k,1]]]], puntos[[segmentos[[k,2]]]]}, t,
centro, sol}, centro = {Xmin + hxInt*(j - 0.5), Ymax - hyInt*(i - 0.5)};
sol = Minimize[{NormCell[s[[1]] + t*(s[[2]] - s[[1]]) - centro],
0 <= t <= 1}, {t}][[1]]; sol = sol <= 1; sol];
DentroPointwrtSeg[p_, k_] := Module[{n, n12, p1, p2, sol},
p1 = p - puntos[[segmentos[[k,1]]]]; p2 = p - puntos[[segmentos[[k,2]]]];
n12 = p1 - p2; n = {-n12[[2]], n12[[1]]}; Which[p1 . n12 < 0,
sol = PuntosSalientes[[segmentos[[k,1]]]]; sol = !sol, p2 . n12 > 0,
sol = PuntosSalientes[[segmentos[[k,2]]]]; sol = !sol, True,
sol = p1 . n >= 0]; sol];
DentroCellwrtSeg[i_, j_, listSeg_] :=
Module[{c1 = {Xmin + hxInt*(j - 1), Ymax - hyInt*i}, distancias, k, sol},
distancias = Table[DPSegm[c1, listSeg[[k]], {k, Length[listSeg]}];
sol = Min[distancias]; sol = First[Flatten[Position[distancias, sol]]];
sol = listSeg[[sol]]; sol = DentroPointwrtSeg[c1, sol]; sol];
Dentro[p_] := Module[{distancias, listSeg, i, j, k, sol},
j = Quotient[p[[1]] - Xmin, hxInt] + 1;
i = N2 - Quotient[p[[2]] - Ymin, hyInt];
If[1 <= i <= Length[CeldasInt] && 1 <= j <= Length[CeldasInt[[1]]],
listSeg = CeldasInt[[i,j]]; If[Length[listSeg] > 0,
listSeg = Table[First[listSeg[[k]], {k, Length[listSeg]}];
distancias = Table[DPSegm[p, listSeg[[k]], {k, Length[listSeg]}];
sol = Min[distancias]; sol = First[Flatten[Position[distancias,
sol]]]; sol = listSeg[[sol]]; sol = DentroPointwrtSeg[p, sol],
sol = listSeg], sol = False]; sol];
ActCeldasBC[i1_] := Module[{p, i, j}, p = Most[Esferas[[i1]]]; p = Mean[p];
j = Quotient[p[[1]] - Xmin, hxBC] + 1;
i = N2BC - Quotient[p[[2]] - Ymin, hyBC];
CeldasBC = Insert[CeldasBC, i1, {i, j, 1}]; ];
Vecinas[i1_, d_] := Module[{p, i, j, i2, j2, sol},
p = Most[Esferas[[i1]]]; p = Mean[p];
j = Quotient[p[[1]] - Xmin, hxBC] + 1;
i = N2BC - Quotient[p[[2]] - Ymin, hyBC];
sol = Table[CeldasBC[[i2,j2]], {i2, Max[1, i - d], Min[N2BC, i + d]},
{j2, Max[1, j - d], Min[N1BC, j + d]}]; sol = Flatten[sol];
sol = DeleteCases[sol, i1]; sol];
Admisible[esf_, ind1_, ind2_] := Module[{p, r, a, vec, i, j, i2, j2,
cont = 1, sol}, p = Most[esf]; p = Mean[p]; r = Last[esf];
a = Last[esf]; j = Quotient[p[[1]] - Xmin, hxBC] + 1;
i = N2BC - Quotient[p[[2]] - Ymin, hyBC]; sol = Dentro[p];
If[sol, vec = Table[CeldasBC[[i2,j2]], {i2, Max[1, i - 1],
Min[N2BC, i + 1]}, {j2, Max[1, j - 1], Min[N1BC, j + 1]}];
vec = Flatten[vec]; vec = DeleteCases[vec, ind1];
vec = DeleteCases[vec, ind2]; While[cont <= Length[vec] && sol,
sol = sol && !Corta[esf, Esferas[[vec[[cont]]]]];
cont = cont + 1; ]; sol]; BorrarOutput :=
Module[{nb}, nb = SelectedNotebook[]; SelectionMove[nb, Next,
GeneratedCell]; SelectionMove[nb, All, Cell]; NotebookDelete[nb]; ];
DentroCuad[cuad_, p_] := Module[{i, n, puntos = VertCuad[cuad], sol = True},
For[i = 1, i <= 4, i++, n = puntos[[i]] - puntos[[Mod[i, 4] + 1]];
n = {-n[[2]], n[[1]]}; sol = sol && (p - puntos[[i]]) . n < 0]; sol];
CrearMalla[xmin_, xmax_, ymin_, ymax_, Nx_, Ny_] :=
Module[{i, j, sol}, sol = Table[{xmin + i*N[(xmax - xmin)/Nx],

```

```

    ymin + j*N[(ymax - ymin)/Ny]], {i, 0, Nx}, {j, 0, Ny}];
    sol = Flatten[sol]; sol = Partition[sol, 2]; sol];
    GenPart[r_, _] := Module[{contr = 0.1, i, sol},
    sol = {{-r + contr*r, 0}, {r - contr*r, 0}};
    sol = Table[sol[[i]] . {{Cos[ ], Sin[ ]}, {-Sin[ ], Cos[ ]}},
    {i, Length[sol]}]; AppendTo[sol, r]; sol];
    Corta[p1_, p2_] := Module[{ = 0.00125, i, j, A2, sol = False},
    q1 = {p1[[1]], p1[[2]], (1 - )*p1[[3]]};
    q1 = {p1[[1]], p1[[2]], (1 - )*p1[[3]]};
    q2 = {p2[[1]], p2[[2]], (1 - )*p2[[3]]};
    For[i = 1, i <= Length[q1] - 1, i++, For[j = 1, j <= Length[q2] - 1,
    j++, sol = sol || (q1[[i]] - q2[[j]]) . (q1[[i]] - q2[[j]]) <
    (Last[q1] + Last[q2])^2; If[sol, Break[]]; If[sol, Break[]];
    sol];
    Centro[i_, j_, r_, a_] := Module[{x, y, p1 = Esferas[[i]], p2 = Esferas[[j]],
    p3 = GenPart[r, a], A1 = {}, A2 = {}, i1, j1, c, sol = {}},
    For[i1 = 1, i1 <= Length[p1] - 1, i1++, For[j1 = 1, j1 <= Length[p3] - 1,
    j1++, AppendTo[A1, {p1[[i1]] - p3[[j1]], Last[p1] + Last[p3]}]]];
    For[i1 = 1, i1 <= Length[p2] - 1, i1++, For[j1 = 1,
    j1 <= Length[p3] - 1, j1++, AppendTo[A2, {p2[[i1]] - p3[[j1]],
    Last[p2] + Last[p3]}]]]; For[i1 = 1, i1 <= Length[A1], i1++,
    For[j1 = 1, j1 <= Length[A2], j1++,
    c = {x, y} /. Solve[{(x, y) - A1[[i1,1]]} . ((x, y) - A1[[i1,1]]) ==
    A1[[i1,2]]^2, ((x, y) - A2[[j1,1]]) . ((x, y) - A2[[j1,1]]) ==
    A2[[j1,2]]^2}, {x, y}]; sol = Join[sol, c]; ];
    sol = Select[sol, #1 Reals & ];
    sol = Table[{p3[[1]] + sol[[i1]], p3[[2]] + sol[[i1]], p3[[3]]},
    {i1, Length[sol]}]; sol = Select[sol,
    !Corta[#1, p1] && !Corta[#1, p2] & ]; sol = Union[sol]; sol];
    Centroide[p_] := Module[{sol = Most[p]}, sol = Mean[sol]; sol];

```

Graficación

```

    PlotPuntos := ListPlot[puntos, PlotStyle -> PointSize[0.02],
    DisplayFunction -> Identity];
    PlotSegm := Graphics[Table[{Thickness[0.006], Hue[1],
    Line[{puntos[[segmentos[[i,1]]]], puntos[[segmentos[[i,2]]]]}],
    {i, Length[segmentos]}]];
    PlotEtiquetas := Module[{i, sol}, sol = {}; For[i = 1, i <= Length[puntos],
    i++, AppendTo[sol, Text[ToString[i], puntos[[i]], {0, -1}]]];
    For[i = 1, i <= Length[segmentos], i++, AppendTo[sol,
    Text[ToString[i], (puntos[[segmentos[[i,1]]]] +
    puntos[[segmentos[[i,2]]]])/2, {0, -1}]]];
    For[i = 1, i <= Length[Esferas], i++, AppendTo[sol,
    Text[ToString[i], Centroid[Esferas[[i]]]]]; sol = Graphics[sol];
    sol]; PlotCeldasInt :=
    Graphics[Join[Table[Line[{{Xmin + i*hxInt, Ymin}, {Xmin + i*hxInt,
    Ymax}}], {i, 0, N1}], Table[Line[{{Xmin, Ymin + i*hyInt},
    {Xmax, Ymin + i*hyInt}}], {i, 0, N2}]]];
    PlotCeldasBC := Graphics[
    Join[Table[Line[{{Xmin + i*hxBc, Ymin}, {Xmin + i*hxBc, Ymax}}],
    {i, 0, N1BC}], Table[Line[{{Xmin, Ymin + i*hyBC},
    {Xmax, Ymin + i*hyBC}}], {i, 0, N2BC}]]];
    PlotPart[i_] := Module[{j, color, sol}, color = Random[];
    sol = Table[{Hue[color], Disk[Esferas[[i,j]], Last[Esferas[[i]]]]},
    {j, Length[Esferas[[i]]] - 1}]; sol = Graphics[sol]; sol];
    PlotEsferas := Module[{i, j, sol},

```

```

sol = Table[PlotPart[i], {i, Length[Esferas]}; sol];
GrafCentrosBC[i_, j_] := Module[{k, sol = {}},
  For[k = 1, k <= Length[Esferas], k++, If[Count[CeldasBC[[i,j]], k] > 0,
    AppendTo[sol, k]]; sol = Table[{Hue[1], PointSize[0.02],
    Point[Most[Esferas[[sol[[k]]]]]}, {k, Length[sol]};
  sol = Graphics[sol]; sol]; PlotPuntoInic :=
Graphics[{PointSize[0.02], Point[puntoInicial]};
GrafTodo := Show[PlotEsferas, PlotSegm, PlotPuntoInic, PlotEtiquetas,
  AspectRatio -> Automatic, DisplayFunction -> $DisplayFunction,
  Axes -> False, PlotRange -> All];

```

Programa Principal

```

Esferas = {}; esf = GenPart[rAl, 0];
esf = {esf[[1]] - esf[[2]] - {esf[[3]], 0} + puntoInicial,
  -{esf[[3]], 0} + puntoInicial, Last[esf]}; AppendTo[Esferas, esf];
esf = GenPart[rAl, 0]; esf = {{esf[[3]], 0} + puntoInicial,
  esf[[2]] - esf[[1]] + {esf[[3]], 0} + puntoInicial, Last[esf]};
AppendTo[Esferas, esf];
For[i = 1, i <= 2, i++, ActCeldasBC[i]; CapExt = {1, 2};
While[Length[CapExt] > 0, Print[Length[Esferas]]; BorrarOutput;
  piv = Last[CapExt]; vecinas = Vecinas[piv, 2]; aumento = 0;
  For[i = 1, i <= Length[vecinas], i++, r = rAl; a = aAl;
    nesf = Centro[piv, vecinas[[i]], r, a]; While[Length[nesf] > 0,
      If[Admisible[Last[nesf], piv, vecinas[[i]]], Break[]];
      nesf = Most[nesf]; If[Length[nesf] > 0, nesf = Last[nesf];
      aumento = aumento + 1; AppendTo[Esferas, nesf];
      AppendTo[vecinas, Length[Esferas]]; AppendTo[CapExt,
        Length[Esferas]]; ActCeldasBC[Length[Esferas]]; Break[]]; ];
If[aumento == 0, CapExt = Most[CapExt]]; ];

```

Programa para la construcción de una partícula formada por colisión de dos discos tangente a otras dos.

Constantes

```

r1 = 70.8393; c11 = {8.54485, -41.2608}; c12 = {-21.1669, 39.0853};
r2 = 44.392; c21 = {138.917, 65.1216}; c22 = {194.364, 75.2494};
r3 = 33.0386; c031 = {169.592, -127.522}; c032 = {193.025, -89.052};
sol1 = {445.032, -259.936}; sol2 = {468.465, -221.466}; LisC = {};
MatC = {{c11, c12}, {c21, c22}}; MatC0 = {c031, c032};

```

Funciones

```

Off[Solve::ifun]; IntCirc[c1_, r1_, c2_, r2_] :=
Module[{sol}, sol = Solve[{Norm[{x, y} - c1] == r1, Norm[{x, y} - c2] ==
  r2}, {x, y}]; If[sol != {}, sol = {x, y} /. sol]; sol]

```

Cálculo

```

i = 1; j = 1; k = 1; l = 1; For[i = 1, i <= 2, i++,
  For[j = 1, j <= 2, j++, For[k = 1, k <= 2, k++, For[l = 1, l <= 2, l++,
    sol = IntCirc[MatC[[1,i]] - MatC0[[j]], r1 + r3,
      MatC[[2,k]] - MatC0[[l]], r2 + r3]; LisC = Join[LisC, sol]; ]]]];

```

Graficación

```

gruesoLinea = 0.005; Off[General::spell1];
Plot1 = Graphics[{Thickness[gruesoLinea], Hue[0.5], Circle[c11, r1]},
  {Hue[0.5], Thickness[gruesoLinea], Circle[c12, r1]}];
Plot2 = Graphics[{Hue[0.3], Thickness[gruesoLinea], Circle[c21, r2]},
  {Hue[0.3], Thickness[gruesoLinea], Circle[c22, r2]}];
Plot3 = Graphics[{Hue[0], Thickness[gruesoLinea], Circle[c031, r3]},
  {Hue[0], Thickness[gruesoLinea], Circle[c032, r3]}];

```

```

For[i = 1, i <= Length[LisC], i++,
  PlotSol = Graphics[{{Hue[0], Thickness[gruesoLinea],
    Circle[LisC[[i]] + c031, r3]}, {Hue[0], Thickness[gruesoLinea],
    Circle[LisC[[i]] + c032, r3]}}]; Show[Plot1, Plot2, Plot3, PlotSol,
  AspectRatio -> Automatic, PlotRange -> All]; ]

```

Programa para la construcción de una partícula formada por colisión de dos esferas tangente a otras tres.

Paquetes a usar

```

<< "Geometry`Polytopes`"; << "Graphics`Shapes`"; << "Graphics`Graphics3D`";
Constantes

```

```

vert = N[Join[Vertices[Triangle], {{1.4, -1.4}}]];
vert = Table[Append[vert[[i]], 0], {i, Length[vert]}]; cont = 0.8;

```

Funciones

```

Off[Solve::ifun]; rAl := Random[Real, {0.125, 0.5}];
IntEsf[i_, j_, k_] := Module[{esf1 = A1[[i]], esf2 = A2[[j]],
  esf3 = A3[[k]], p, sol}, Clear[x, y, z]; p = {x, y, z};
  sol = p /. NMinimize[{(p - Most[esf1]) . (p - Most[esf1]) -
    Last[esf1]^2)^2 + ((p - Most[esf2]) . (p - Most[esf2]) -
    Last[esf2]^2)^2 + ((p - Most[esf3]) . (p - Most[esf3]) -
    Last[esf3]^2)^2, p][[2]]; sol];

```

Construcción de las partículas iniciales

```

p1 = {{-0.46995558149054784, 1.1107751337991965, 0, 0.27398544033108274},
  {-0.5300444185094522, 0.6212756737696806, 0, 0.27398544033108274}};
p2 = {{-0.3536404043691338, -0.5349159693738716, 0, 0.40223847233735804},
  {-0.6463595956308662, -1.1971348381950055, 0, 0.40223847233735804}};
p3 = {{0.9290448770110105, 0.3796785511518203, 0, 0.42916863600141913},
  {1.0709551229889895, -0.3796785511518203, 0, 0.42916863600141913}};
p4 = {{1.2430458766652794, -1.4736657479107889, 0, 0.384},
  {1.5569541233347204, -1.326334252089211, 0, 0.384}};
part = {p1, p2, p3, p4}; A1 = Table[Join[Most[p1[[i]]] - Most[p4[[j]]],
  {Last[p1[[i]]] + Last[p4[[j]]}], {i, 2}, {j, 2}];
A1 = Partition[Flatten[A1], 4];
A2 = Table[Join[Most[p2[[i]]] - Most[p4[[j]]],
  {Last[p2[[i]]] + Last[p4[[j]]}], {i, 2}, {j, 2}];
A2 = Partition[Flatten[A2], 4];
A3 = Table[Join[Most[p3[[i]]] - Most[p4[[j]]],
  {Last[p3[[i]]] + Last[p4[[j]]}], {i, 2}, {j, 2}];
A3 = Partition[Flatten[A3], 4];

```

Graficación

```

Off[General::spell1]; PlotVert := Graphics3D[Point /@ vert];
PlotPart := Module[{i, sol},
  sol = Join[Table[ParametricPlot3D[part[[i,1,4]]*{Cos[t]*Cos[u],
    Sin[t]*Cos[u], Sin[u]} + Most[part[[i,1]]], {t, 0, 2*Pi},
    {u, -Pi/2, Pi/2}, DisplayFunction -> Identity,
    ViewPoint -> {0, 0, 2}], {i, Length[part]}],
  Table[ParametricPlot3D[part[[i,1,4]]*{Cos[t]*Cos[u], Sin[t]*Cos[u],
    Sin[u]} + Most[part[[i,2]]], {t, 0, 2*Pi}, {u, -Pi/2, Pi/2},
    DisplayFunction -> Identity], {i, Length[part]}]]; sol];
Plotp5 := Module[{i, sol},
  sol = Table[ParametricPlot3D[p5[[i,4]]*{Cos[t]*Cos[u], Sin[t]*Cos[u],
    Sin[u]} + Most[p5[[i]]], {t, 0, 2*Pi}, {u, -Pi/2, Pi/2},
    DisplayFunction -> Identity, ViewPoint -> {0, 0, 2}], {i, 2}]; sol];
GrafTodo := Show[PlotVert, PlotPart, Plotp5, AspectRatio -> Automatic,
  PlotRange -> All, DisplayFunction -> $DisplayFunction,

```

```

ViewPoint -> {0, 0, 2}];
Cálculo de la partícula tangente
centros = {}; For[i = 1, i <= 2, i++, For[j = 1, j <= 2, j++,
  For[k = 1, k <= 2, k++, AppendTo[centros, IntEsf[i, j, k]]; ]]];
centros = Select[centros, #1 Reals & ];
For[i = 1, i <= Length[centros], i++, c = centros[[i]];
  p5 = {Join[Most[p4[[1]]] + c, {Last[p4[[1]]}], Join[Most[p4[[2]]] + c,
    {Last[p4[[2]]}]]}; Print[p1]; Print[p2]; Print[p3]; Print[p4];
  GrafTodo; ]

```

Programa para aproximar una elipse mediante circunferencias.

```

a = 2; b = 1; c = Sqrt[a^2 - b^2]; radios = {}; N1 = 3; h = c/N1;
centros = Table[N[i*h], {i, 0, N1}];
centros = Union[Join[centros, -centros]];
centros = Table[{centros[[i]], 0}, {i, Length[centros]}];
radios = Table[NMinimize[{Norm[{x, y} - centros[[i]],
  x^2/a^2 + y^2/b^2 == 1}, {x, y}][[1]], {i, Length[centros]}];
PlotCentros := Module[{i, sol},
  sol = Graphics[Table[{PointSize[0.02], Point[centros[[i]]],
    {i, Length[centros]}]]; sol]; PlotCircunf :=
Module[{i, sol}, sol = Graphics[Table[Circle[centros[[i]], radios[[i]],
  {i, Length[centros]}]]; sol]; PlotElipse :=
ParametricPlot[{a*Cos[t], b*Sin[t]}, {t, 0, 2*Pi},
  PlotStyle -> {RGBColor[0, 0, 1]}, DisplayFunction -> Identity];
GrafTodo := Show[PlotCircunf, PlotCentros, PlotElipse,
  AspectRatio -> Automatic];

```