

MODELAMIENTO DE ESTRUCTURAS DE *BIMSOILS* MEDIANTE EL EMPAQUETADO DE PARTÍCULAS CIRCULARES EN \mathbb{R}^2 ^a

MODELING OF BIMSOILS STRUCTURES BY MEANS OF CIRCULAR PARTICLES PACKED IN \mathbb{R}^2

SUÁREZ-BURGOA, LUDGER O. ^{b *}, ARIZA-TRIANA, ANDRÉS ^c, MONTOYA-ARAQUE
EXNEYDER A. ^d

Recibido 22-05-2018, aceptado 22-04-2019, versión final 27-06-2019.

Artículo Investigación

RESUMEN: El presente artículo explica el algoritmo y la implementación que se plantea para generar de forma automática el empaquetamiento de círculos de diferentes diámetros que se acomodan de forma tangencial en cada triángulo de un arreglo de triángulos que cubren un polígono cerrado \mathbb{R}^2 . La manera de generar este empaquetamiento es recursivo a varios niveles de profundidad, lo que genera partículas de diferentes diámetros que tiende al modelo a ser autosimilar para el intervalo de partículas más pequeñas y se desvía para partículas grandes. Sin embargo, se logra autosimilitud en los diámetros apagando o borrando algunas partículas pequeñas. La autosimilitud así lograda es dependiente de la forma del polígono cerrado. Cuando dicho polígono cerrado representa el marco geométrico de un talud en \mathbb{R}^2 (una forma de polígono cerrado especial y tradicional para este tipo de problemas geotécnicos) se generan partículas autosimilares para representar geomateriales compuestos por bloques de roca embebidos en una matriz de suelo (*bimsoils*, acrónimo del Inglés *blocks-in-matrix soils*). Con este modelo geométrico generado es posible analizar la estabilidad de taludes mediante el método de equilibrio límite o mediante un análisis de esfuerzo-deformación. El algoritmo y la implementación está asociados al desarrollo de un software de aplicación denominado *circpacker*, el cual está escrito en PYTHON3[®] y se lo presenta a la comunidad como un código abierto y libre.

PALABRAS CLAVE: bloques de rocas en matriz de suelo; *bimsoil*; estabilidad de taludes; empaquetamiento de círculos; fractal; Python; software de aplicación.

^aSuárez-Burgoa, Ludger O., Ariza-Triana, Andrés & Montoya-Araque Exneyder A. (2019). Modelamiento de estructuras de *bimsoils* mediante el empaquetado de partículas circulares en \mathbb{R}^2 . *Rev. Fac. Cienc.*, 8 (2), 115–137. DOI: <https://doi.org/10.15446/rev.fac.cienc.v8n2.72343>

^bPh. D., Profesor Asociado, Universidad Nacional de Colombia, Medellín

* Autor para correspondencia: losuarezb@unal.edu.co

^cCiv. Eng., Grupo de Investigación de Geotecnia, Universidad Nacional de Colombia, Medellín

^dM. Eng., Grupo de Investigación de Geotecnia, Universidad Nacional de Colombia, Medellín

ABSTRACT: This article explains the algorithm and its implementation that are proposed to generate automatically the packing of circles of different diameters in each triangle of a triangular arrangement that covers a closed polygon in \mathbb{R}^2 . The manner to generate this packing is recursive at various level of depth, that generates particles of different diameters that tend to be self-similar model for the interval of small particles, and they divert for big diameters. However, it is attained a self-similarity when switching-off or deleting some of the small particles. The self-similarity so reached is dependent on the shape of the closed polygon. When the mentioned closed polygon represents the geometric frame for a slope in \mathbb{R}^2 (a closed polygon shape special and traditional for this kind of geotechnical problems) it possible generate particles (self-similar in diameters) to represent geomaterials composed by rock blocks in a matrix of soil (*bimsoils*). With this generated geometric model, it is possible to analyze the stability of slopes by the limit equilibrium method or by the stress–deformation analysis. The algorithm and the implementation are associated with the applied software called *circpacker*, which is written in PYTHON3[®] and it is presented to the community as an open and free software.

KEYWORDS: Blocks–in–matrix soil; *bimsoil*; slope stability; circle packing; fractal; Python; application software.

1. INTRODUCCIÓN

Desde hace al menos medio siglo se han planteado alternativas al problema de la acomodación de ciertas figuras planas congruentes sin traslaparse y cubriendo la mayor fracción posible de una cierta región en el plano (*e.g.* Kausch-Blecken, H. H. *et al* (1970)). El caso más sencillo de figura plana es el círculo (*i.e.* el disco) o desde el punto de vista físico la *partícula circular*. La generación de partículas circulares dentro de un contorno cerrado en el plano (espacio \mathbb{R}^2) es una aplicación ampliamente extensible, desde fines artísticos hasta aplicaciones ingenieriles.

El presente artículo se centra en brindar a los lectores una herramienta computacional libre para empaquetar partículas circulares en un polígono cerrado en el plano, *i.e.* en \mathbb{R}^2 . Esto con el fin de enfocar su aplicación en la generación de modelos teóricos de un material geológico (sin denotación genética) denominado *bimsoil* (del acrónimo en Inglés de *Blocks-in-matrix soil*), definido como una *mistura geológica* de bloques rocosos geotécnicamente significantes embebidos en una matriz de textura más fina, en este caso de suelo, o igualmente rocosa en el caso de un *bimrock* (Medley, E., 1994).

Tanto los *bimsoils* como los *bimrocks* poseen dos condiciones por definición. La primera condición indica que entre la matriz y los bloques debe existir un contraste mecánico suficiente de tal forma que una eventual superficie de falla se desarrollaría alrededor de los bloques y no a través de ellos, definiendo así, una geometría denominada *tortuosa* (Medley, E., 1994; Medley, E., 2004). En Kalender, A. *et al* (2014) se compilaron los criterios cuantitativos actualmente aceptados para definir que el contraste mecánico es suficiente para desarrollar la tortuosidad de la superficie de falla; estos criterios comparan el módulo elástico, el ángulo de fricción interna y/o la resistencia a la compresión uniaxial entre los bloques y la matriz.

Por otra parte, la segunda condición indica que la distribución del tamaño de los bloques tiene un com-

portamiento *fractal* según las observaciones realizadas por Medley, E. W. & Lindquist, E. (1995) mientras estudiaba la independencia de escala del tamaño de bloques en un melánge que lo bautizo bajo el nombre de bimrock (*i.e.* definición de *autosimilaridad*).

Un *fractal* es un sistema autosimilar; es decir, que está compuesto por elementos que son réplicas a diferentes escalas de una muestra paterna inicial (Tanton, J., 2005). Para saber si la distribución de los *tamaños de bloques* es autosimilar, esta variable se analiza a diferentes escalas de observación, relacionando en este caso, los diferentes tamaños *versus* la frecuencia de los mismos en un histograma (tomando como variable independiente el tamaño de los bloques).

A partir de dicho histograma se toman las tuplas que resultan de la discretización de los intervalos y se desarrolla una correlación lineal en el espacio $\log - \log$ (*histogramas logarítmicos* o simplemente *log-histograms*). De este modo, la pendiente de la correlación lineal que se obtiene (dada en valor absoluto) se denomina *coeficiente fractal*, se denota con D y es

$$D = \frac{\log N_r}{\log r}; \quad (1)$$

donde N_r es la frecuencia de la variable r , siendo r el valor medio del intervalo de clase de una variable que representa el diámetro, el área, el volumen o el peso de los bloques asumidos circulares o esféricos.

La herramienta desarrollada que se presenta en este artículo corresponde a un programa computacional aplicado y propio que permite modelar la distribución de bloques circulares de diferentes tamaños dentro de cualquier polígono cerrado (*i.e.* un contorno convexo o cóncavo) en el plano, tratando de lograr lo más que se pueda a que se cumpla la condición de la autosimilitud previamente descrita.

Los modelos así logrados pueden usarse para representar la estructura geométrica de los *bimsoils* en un talud (*i.e.* tener bloques circulares en una matriz), los cuales se puede usar para el análisis de estabilidad modelando automáticamente y sin subjetividad ni sesgo, cuando se compara a la generación de estructuras geométricas creadas a criterio del analista.

Un modelo tal se ha usado en la estabilidad de taludes en dos dimensiones, donde a partir de él se ha generado las *superficies de falla tortuosas* por medio del algoritmo A^* aplicado por Montoya-Araque, E. A. & Suarez-Burgoa (2019).

2. ESTUDIOS EN BIMSOILS Y LA APLICABILIDAD DE LA HERRAMIENTA DESARROLLADA

El estudio en *bimsoils* y *bimrocks* inició con los trabajos seminales de Lindquist, E. (1994) y Medley, E. (1994) hace ya 25 años, pero parece que apenas hace unos 10 años hubo una explosión de publicaciones de

artículos referente al tema.

Es de especial interés el presente desarrollo computacional (*i.e.* empaquetamiento de partículas circulares tangentes en \mathbb{R}^2) para su aplicación en temas de análisis de estabilidad de taludes, ya que como ha sido sintetizado por Montoya-Araque, E. A. (2019): los estudios realizados por Irfan, T. Y. & Tang, K. Y. (1993), Medley, E. W. & Sanz, P. F. (2004), Kim, C. *et al* (2004), Guerra, C. & Pinzón, J. (2015), Minuto, D. & Morandi, L. (2015), Napoli, M. L. *et al* (2018), Khorasani, E. *et al* (2019), Khorasani, E. *et al* (2019) se desarrollaron con modelos teóricos de taludes compuestos por *bimsoils* generando de forma heurística la cantidad, tamaño y distribución de bloques dentro de un polígono cerrado.

A partir de un modelo de la estructura geométrica, es posible definir automáticamente superficies de falla tortuosas mediante el procedimiento explicado en Montoya-Araque, E. A. & Suarez-Burgoa (2019), por ejemplo. Por tanto, se convierte en una interesante combinación de recursos computacionales para una mejor práctica de esta disciplina, y adicionalmente, para otras donde se involucren a los *bimsoils* y/o *bimrocks*.

Las siguientes secciones se centran en el desarrollo propiamente de la herramienta computacional que permite la modelación de estructuras de *bimsoils* mediante el empaquetamiento de partículas circulares dentro de polígonos en \mathbb{R}^2 .

3. CÍRCULOS QUE LLENAN FIGURAS PLANAS

El problema de llenar un contorno cerrado cualquiera en el plano (en \mathbb{R}^2) mediante círculos tangentes entre sí ha sido estudiado por los matemáticos al menos desde la séptima década del siglo XX. Hoy en día el problema así planteado se define con el nombre de *empaquetamiento osculatorio* (del Inglés *osculatory packing*; el término *oscular* se refiere a un toque tangencial, del latín *osculari* que significa *besar*).

En este artículo se plantea dos casos: el empaquetamiento osculatorio en un círculo y el empaquetamiento osculatorio en cualquier triángulo.

El caso del empaquetamiento en el círculo es común, mientras que para el triángulo irregular no lo es; pero el empaquetar en un triángulo es de gran interés porque si se logra esto, es posible cubrir espacios irregulares en \mathbb{R}^2 , porque cualquier subconjunto de \mathbb{R}^2 puede ser cubierto por una malla de triángulos irregulares.

3.1. Círculos que llenan un círculo

El *tamiz de Apolonio* (*i.e.* *Apollonian Gasket* en Inglés, llamado también el empaquetamiento de Leibnitz y que lo nombraremos con la sigla TA) es el objeto geométrico resultado de aplicar un algoritmo que tiene el fin de llenar un círculo externo mayor con pequeños círculos menores, de tal forma de que cada espacio que

se cree una vez se haya añadido un círculo sea llenado con otro círculo tal que este último sea tangente a los círculos ya existentes.

El algoritmo es recursivo y es infinito, por tanto se debe indicar la *profundidad de la recursividad*. La profundidad de la recursividad es un número entero que indica cuántas veces se va a repetir el algoritmo en un mismo objeto, es un concepto de fractales. Por ejemplo, una profundidad $n = 3$ indica que la recursividad se debe repetir tres veces.

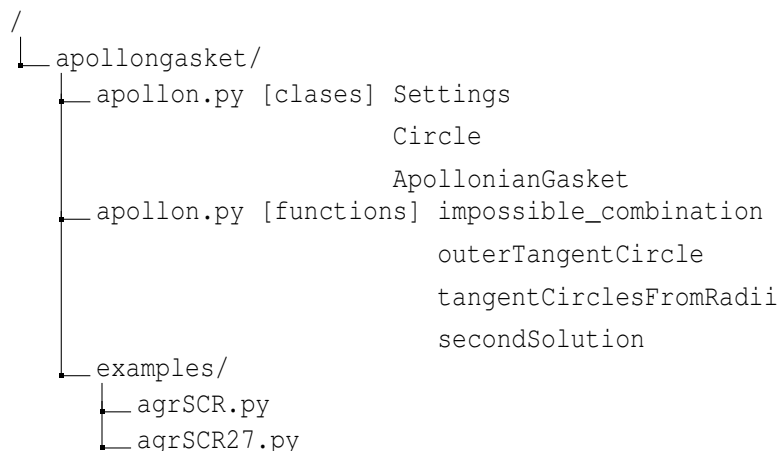
En el AP se obtienen diversos resultados dependiendo de la posición y el diámetro de los primeros dos círculos (hasta tres) que se introducen dentro del círculo externo mayor. El número de círculos (N_c) que pueden entrar en el círculo del contorno en el TA en función de la profundidad n está dado por:

$$N_c = 2 \times 3^{n+1}. \quad (2)$$

A continuación se verá cómo se genera un TA y cómo se calcula su dimensión fractal siguiendo el proceso de cálculo experimental y no analítico. Para ello se usó una versión adaptada del programa aplicado abierto `apollon.py` de Sandig (Sandig, L. , 2014) que se han creado para generar TAs. Esta aplicación se puede ejecutar en la nube en la dirección <http://lsandig.org/cgi-bin/apollon/index.cgi> y contiene ahí mismo la explicación de cómo se han generados los AP.

Las variables de partida para generar un TA son tres círculos C_1 , C_2 y C_3 los cuales serán tangentes entre sí. A partir de estos círculos se encuentra un cuarto círculo C_4 que es tangente a los anteriores tres pero que circunscriben a ellos. Este cuarto círculo C_4 es el máximo de todos y es el que define el contorno del cálculo y donde dentro de él se generarán los subsiguientes círculos.

La Figura 1 muestra un TA generado con el programa de Sandig (Sandig, L. , 2014) para los tres primeros círculos con radios respectivos de 0.25 unidades de longitud [L], 0.25 [L] y 0.5 [L], para una profundidad de cinco. El TA generado está inscrito en el círculo mayor C_4 de radio 0.761 [L]; *i.e.* diámetro de 1.522 [L]. El siguiente árbol muestra la estructura del módulo `apollon.py`, utilizado para generar el TA de la Figura 1.



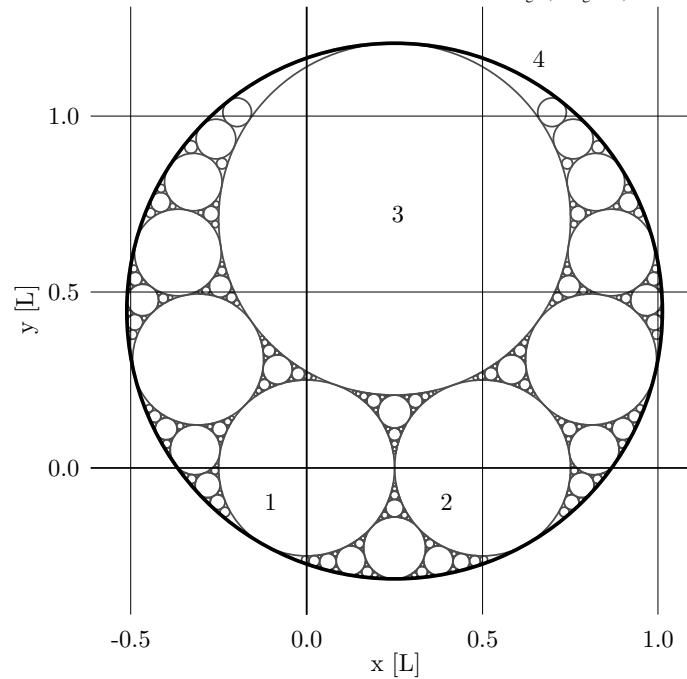


Figura 1: Tamiz de Apolonio tomando como semilla tres círculos; donde dos de ellos tienen radios 0.25 [L] y el tercero 0.5 [L].

Fuente: Elaboración propia con el archivo agSCR.py

Los métodos de la clase `Circle` son: `center`, `radius`, `diameter`, `perimeter`, `areas` y `curvature`. Por su parte, los métodos de la clase `ApollonianGasket` son `recurse` y `generate`.

Como se puede observar, al dar dos círculos con dos radios iguales se logra tener un TA simétrico con el diámetro vertical de C_4 ; además, el centro de todo el TA se da en el centro del primer círculo generado, situación que no es inconveniente porque se puede trasladar todo el TA hacia el centro del círculo C_4 .

A parte de generar el TA, el código fue adaptado para indagar sobre los diámetros generados sin tomar en cuenta el círculo C_4 que define el contorno del objeto. Dentro del círculo que define el contorno del TA, el círculo de mayor diámetro tiene unidades de 1.314[L], solo 86.34% el diámetro del círculo que define el contorno. Asimismo, el círculo de menor diámetro tiene unidades de 0.002[L], un 0.13% el diámetro del círculo que define el contorno. Esto indica que el TA generado abarca muchos diámetros de círculos, tal como se muestra en el histograma de la Figura 2(a); pero aun no se sabe si es autosimilar.

3.1.1. Verificación de la condición de autosimilitud

Para saber si una figura generada por recursividad es autosimilar se tiene que obtener su *coeficiente fractal*. Para algunos fractales se puede deducir de forma analítica este coeficiente, pero no para todos. Por ejem-

plo, para el caso de un fractal de partículas circulares, en la mayoría de las veces no es posible obtener el coeficiente fractal de forma analítica; por tanto, este se estima haciendo una gráfica \log_2 – \log_2 donde en el eje de abscisas se anota el logaritmo (cualquier logaritmo) del valor central de cada uno de los intervalos de diámetros de las partículas circulares; y en el eje de ordenadas, el mismo logaritmo pero de las cantidades presentes de los diámetros en cada intervalo, *i.e.* la frecuencia del intervalo.

El coeficiente fractal entonces será la pendiente (en valor absoluto) de una recta que se ajusta a todos los datos que representan el histograma del tamaño de partículas en la gráfica \log_2 – \log_2 . Además, la recta así encontrada debe ajustarse en todos los datos (no en algunos) y debe ser constante para toda la distribución de tamaños de partículas.

La gráfica \log_2 – \log_2 del ejemplo del TA generado arriba se puede observar en la Figura 2(b). Se puede observar que sí es posible ajustar una recta en los diámetros más pequeños hasta un diámetro de aproximadamente de $2^{-2} \equiv 0.25 [L]$; de ahí en adelante, las partículas están presentes con algún sesgo o carencia que no hace cumplir la autosimilitud. Por tanto, el TA, al menos el generado bajo el procedimiento de arriba, no es autosimilar al considerar todos los círculos generados.

Otras figuras interesantes de empaquetamiento de círculos en círculos son la cadena de Steiner (*Steiner Chain*) y la cadena en el Arbelós.

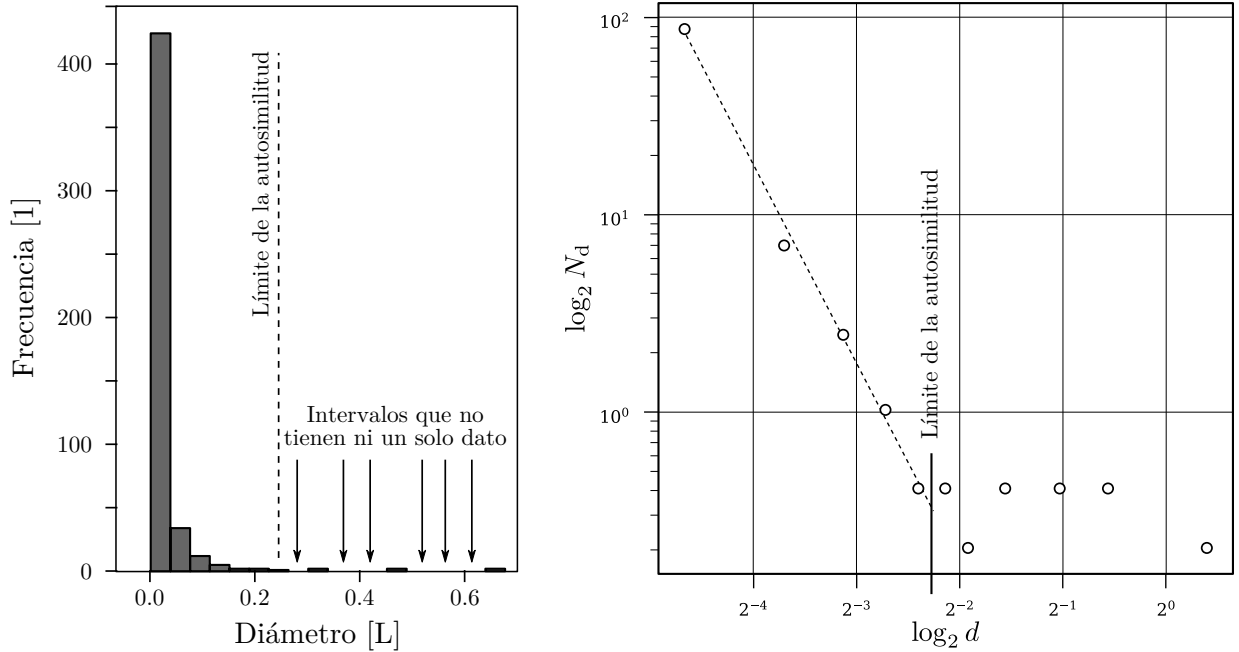
3.2. Círculos que llenan un triángulo

Partiendo de la idea de la forma cómo se genera un TA (visto en la sección anterior) los autores de este artículo crearon un procedimiento para generar círculos tangentes entre sí dentro de un triángulo cualquiera, donde los círculos tangentes van acomodándose en los espacios que sobran una vez se llega a colocar un círculo de forma inscrita en el triángulo (este círculo es el *incírculo del triángulo*).

Si el empaquetamiento dentro del triángulo irregular es de un solo un círculo y además este es el incírculo, se dice que el triángulo fue empaquetado a su máxima relación partícula–área posible; es decir, ninguna otra partícula circular de radio mayor al incírculo puede empaquetarse dentro del triángulo (lo cual resulta obvio).

El motivo de esta necesidad (que el empaquetamiento sea dentro de un triángulo) es que un polígono cualquiera cerrado en el espacio \mathbb{R}^2 sí puede ser completamente llenado con una malla de triángulos irregulares (O'Rourke, J. , 1997); por tanto, si se logra llenar cada uno de los triángulos que llena dicho polígono, es posible también llenar ese mismo espacio con círculos todos tangentes.

El procedimiento podría formar (vea posteriormente las conclusiones de este artículo) un objeto que es un fractal de partículas circulares dentro del triángulo, debido a que cada iteración del procedimiento (que representa una profundidad del posible fractal) crea una partícula menor. Lo que se desconoce es si la dis-



(a) Histograma de la distribución de diámetros del TA).

(b) Gráfica \log_2 - \log_2 de los diámetros y cantidades de las partículas circulares del TA).

Figura 2: Límite de la autosimilitud para el ejemplo del TA. Fuente: Elaboración propia con el archivo agSCR.py .

tribución de tamaño de partículas circulares es autosimilar.

El procedimiento planteado indica que un triángulo cualquiera puede ser llenado mediante círculos de forma infinita sin llegar a colmar todo su espacio si se sigue el siguiente procedimiento (Figura 4).

1. Se genera un triángulo $\mathcal{T}_1 \equiv \triangle V_1 V_2 V_3$ cualquiera en el plano.
2. Al triángulo \mathcal{T}_1 se le encuentra el círculo que lo inscribe C_0 (el *incírculo*). El triángulo y el círculo están relacionados y forman la primera profundidad o la primera generación. El triángulo y el incírculo forman tres lugares que pueden ser llenados por más círculos.
3. Si para cada espacio relacionado a cada uno de los tres vértices de este triángulo se hace lo siguiente.
 - a) Se genera un círculo C_1 que es tangente a los lados $\overline{V_1 V_3}$ y $\overline{V_1 V_2}$ y al círculo anteriormente construido C_0 ; donde el cálculo del radio y la posición de este círculo se hace según el procedimiento de Lucca (Lucca, G. , 2017).
 - 1) Entre los lados $(\overline{V_1 V_3}$ y $\overline{V_1 V_2})$ y los dos círculos (C_1 y C_0) hay dos espacios que se pueden llenar con un círculo cada uno (C_{11} y C_{12}); la solución de cualquiera de estos dos círculos se resuelve mediante el método de Descartes (caso especial y descrito en (Bourke, P., 1992))

- para encontrar un círculo tangente a tres círculos, donde el caso especial dicta que uno de esos círculos tiene radio infinito.
- 2) Con cualquiera de los dos nuevos círculos, *e.g.* con el círculo C_{11} , se crea tres nuevos espacios para crear otros dos círculos que los llamaremos C_{111} , C_{112} y C_{113} . Los dos primeros se resuelven igual que el anterior caso.
 - 3) El tercer círculo se resuelve de tal forma que se debe encontrar un círculo que es tangente a tres radios (distintos de infinito); lo que es posible con la solución de Descartes general (descrito también en Bourke, P. (1992)).
 - 4) Los dos últimos pasos a este se repiten para el caso del círculo C_{12} .
- b) Se repite el procedimiento para los siguientes espacios que van hacia el vértice hasta una profundidad n o hasta un límite de diámetro especificado.
4. Para los vértices V_2 y V_3 del triángulo \mathcal{T}_1 se hace lo mismo.

Los subíndices de cada uno de los círculos se pueden acomodar en un árbol estructurado; por tanto, la búsqueda de cada uno de los círculos generados es posible.

De este modo, se puede generar cuantos círculos se quieran; pero cada vez los círculos serán más pequeños generando un *posible fractal* hasta una profundidad que tiende al infinito (que lo denominaremos *pseudofractal*). A la configuración así obtenida se la ha denominado *cadena de círculos en un triángulo* por Lucca (Lucca, G. , 2017); por tanto, el fractal se puede llamar el *pseudofractal de Lucca*.

El número de círculos que se puede generar dentro del triángulo está dado por la siguiente ecuación

$$N_c = \begin{cases} 3^{(2n-1)} + 1 & \text{para } n > 0 \\ 1, & \text{para } n = 0 \end{cases}; \quad (3)$$

donde nuevamente n es la profundidad del fractal siendo $n \in \{\mathbb{N}, 0\}$.

Si $n \rightarrow \infty$ entonces $N_c = \infty$. La serie $(2n - 1)$ es la serie de los números impares; por tanto, N_c crece potencialmente en la base tres de forma potencial en la serie de los números impares. Esto implica que muy rápidamente N_c alcanzará valores altos en \mathbb{N} que puede llegar a colapsar la memoria del computador con valores de n menores a la decena. Si $n = 7$ entonces $N_c = 2187$, valor suficientemente grande para tener un empaquetamiento de círculos en un triángulo; por tanto, se aconseja para el desarrollo de estos fractales que $n < 5$.

La Figura 5 muestra dos ejemplos: un polígono convexo irregular de seis lados y un polígono cóncavo irregular de ocho lados. Ambos ejemplos son conformados por triángulos dentro de los cuales se empaquetan círculos con el procedimiento anteriormente descrito, y como se esperaba, los polígonos en su totalidad fueron llenados por círculos tangentes.

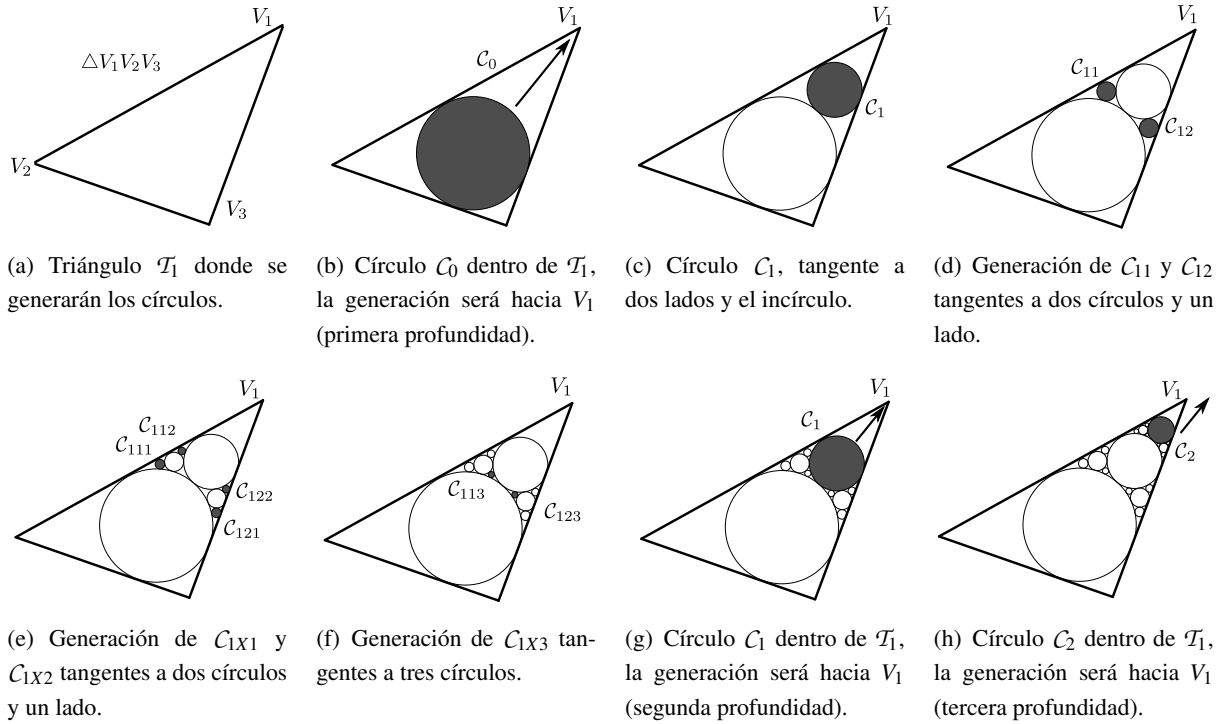


Figura 3: Esquema de la forma de generar círculos tangentes dentro de un triángulo para uno de los tres vértices. Fuente: Elaboración Propia.

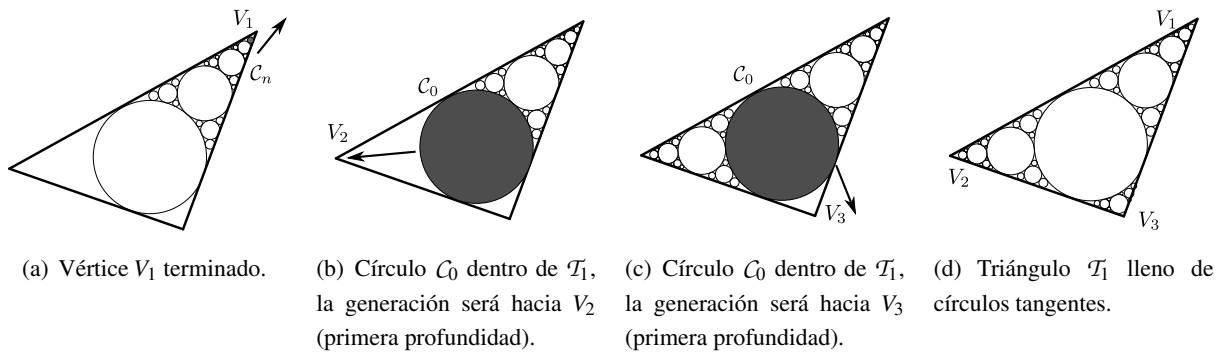
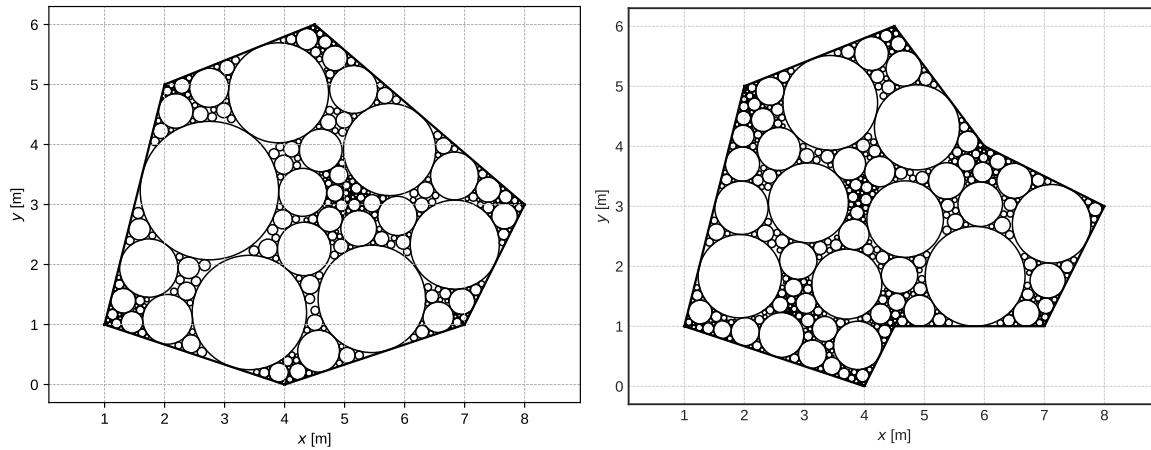


Figura 4: Esquema de la forma de generar círculos tangentes dentro de un triángulo para los restantes vértices hasta llenar el triángulo. Fuente: Elaboración propia.



(a) Polígono irregular de seis lados convexo.

(b) Polígono irregular de ocho lados cóncavo.

Figura 5: Llenado de polígonos en \mathbb{R}^2 con partículas circulares tangentes. Fuente: Elaboración propia.

3.2.1. Verificación de la condición de autosimilitud

Lo que resta ahora para cada uno de los dos ejemplos es verificar si alguno de ellos presenta autosimilitud en la distribución del diámetro de sus partículas. El procedimiento que se hace para esto es el mismo que se usó en el caso de las TAs; es decir, se extrae la información del diámetro de todas las partículas generadas, se elabora un histograma definiendo intervalos de clases, luego para cada clases se obtiene la frecuencia y con ese par de datos se grafica en una escala \log_2 – \log_2 en abscisas el logaritmo del diámetro de las partículas y en ordenadas el logaritmo de la frecuencia.

Si en esa gráfica \log_2 – \log_2 es posible ajustar una recta para toda la extensión de los datos, entonces el desarrollo de partículas en el polígono irregular presenta autosimilitud (dándole el nombre de fractal al algoritmo de construcción); si la recta es posible solo para un dado intervalo, entonces la autosimilitud es parcial (dándole el nombre de pseudofractal al algoritmo de construcción). La Figura 6 muestra los histogramas de los dos primeros ejemplos de la Figura 5.

Se observa que la mayor cantidad de partículas se tiene para los diámetros pequeños y luego van disminuyendo casi en forma exponencial o potencial. Para la Figura 6(a) se tiene pocas cantidades de partículas para los diámetros grandes (*i.e.* mayores a 1.0 [L]), mientras que para la Figura 6(b) se tiene un poco más del doble de partículas grandes y por el orden de los 1.25 [L] a 1.50 [L]; esta situación hace que la cantidad de partículas más pequeñas baje en una cantidad de casi la mitad; es decir de 1200 partículas a 600 partículas aproximadamente.

Si ahora se hace la gráfica \log_2 – \log_2 de esos mismos datos (que se muestra en la Figura 11) se puede ver que la distribución de diámetros es autosimilar parcialmente y no total porque los diámetros mayores no se están ajustando cercanamente a la recta, especialmente para el caso de la Figura 6(b). Por tanto, el algoritmo plan-

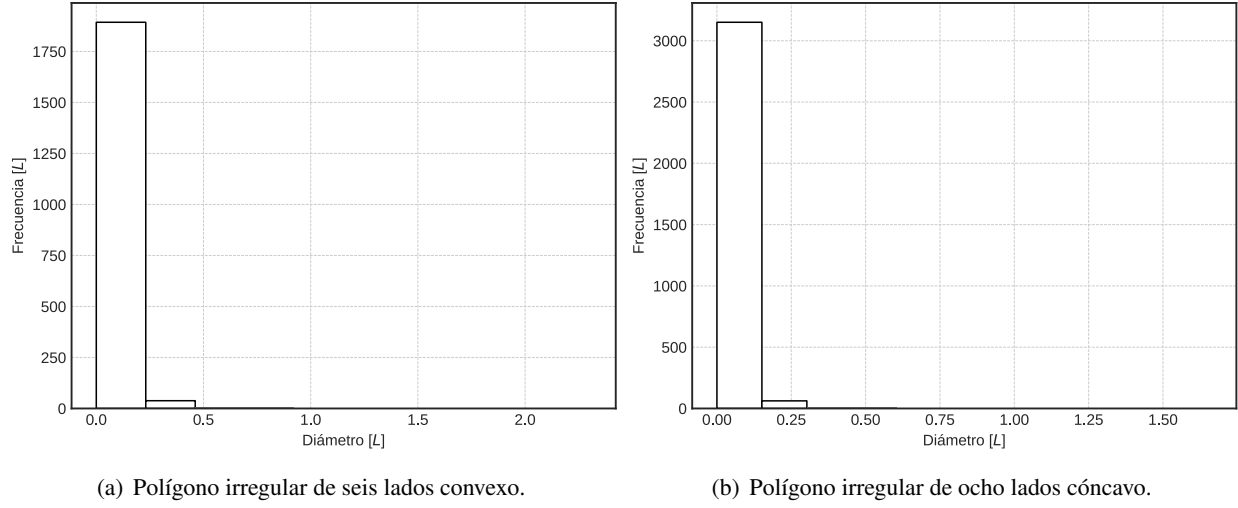


Figura 6: Histogramas de los diámetros de las partículas circulares en los polígonos en \mathbb{R}^2 . Fuente: Elaboración propia.

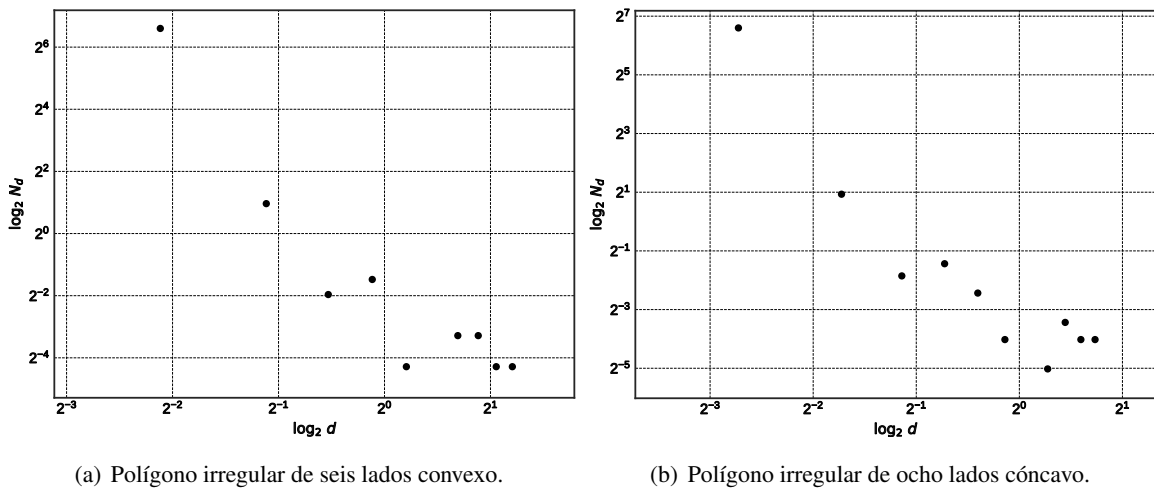


Figura 7: Gráfica \log_2 - \log_2 de los diámetros y cantidades de las partículas circulares en los polígonos en \mathbb{R}^2 . Fuente: Elaboración propia.

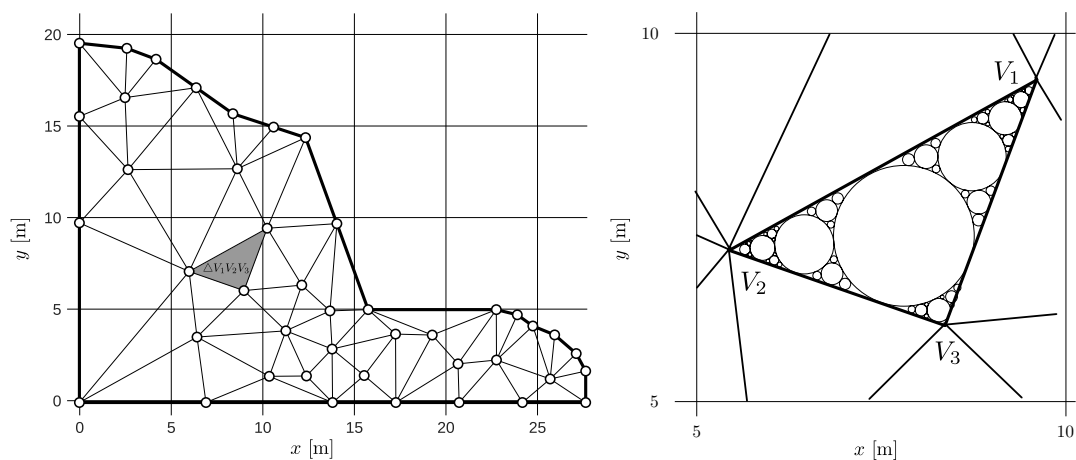
teado, según los términos aquí propuestos, genera pseudofractales, por que si bien tiene un procedimiento recursivo para generarlo, su autosimilitud no es total.

Habría que preguntarse si el cumplir a cabalidad el concepto de fractal bajo la regla de autosimilitud daña o no la aplicación que se le quiere dar a este algoritmo. En principio, no. En algunos *bimsoils* en Colombia (uno de los autores [Suárez-Burgoa]) ha encontrado que la mayoría de estos no son autosimilares para depósitos aluviales y coluviales (observación aún no publicada). La autosimilitud se da más en bimrocks de origen metamórfico dinámico como brechas.

4. SOFTWARE DE APLICACIÓN DESARROLLADO

Para la implementación del procedimiento descrito en la subsección 3.2 se ha escrito en el lenguaje PYTHON3® un software de aplicación (con las siglas SA) denominado *circpacker*. El SA fue desarrollado por uno de los autores del presente artículo (Ariza-Triana).

Este SA tiene la posibilidad de primero crear una malla de triángulos irregulares dentro de cualquier contorno cerrado mediante el método de la *triangulación de Delaunay* codificada en la librería *triangle* (Rufat, D., 2019); sea tanto contornos convexos (*convex hull*) como contornos cóncavos (*concave hull*). Luego, en cada uno de los triángulos se genera las partículas circulares tangentes entre sí, tal como se explica en la Figura 8.

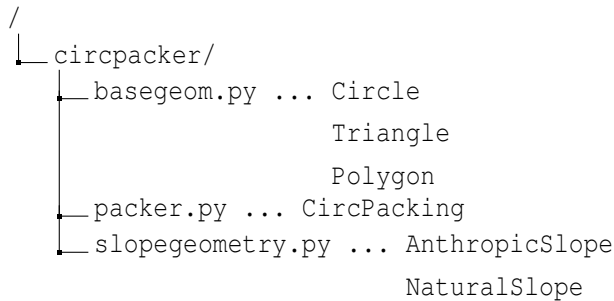


(a) En el polígono irregular se genera una malla de triángulos irregulares. (b) En cada triángulo se genera las partículas circulares.

Figura 8: Forma de generar partículas en un polígono cóncavo que representa un talud. Fuente: Elaboración propia.

El paradigma de programación que se usó para este fin es el de la programación orientada a objetos (OOP)

de la siglas en Inglés de *Object-Oriented Programming* y consta de tres archivos con un total de seis clases. El siguiente árbol muestra la estructura del PA.



En el módulo `basegeom`: la clase `Circle` tiene el método `descartesTheorem`; la clase `Triangle` tiene los métodos `getGeomProperties`, `circInTriangle` y `plot` y la clase `Polygon` tiene los métodos `area` y `plot`. En el módulo `packer`, en la clase `circPacking` se tiene los métodos `triMesh`, `generator`, `plot`, `frecHist` y `logDiagram`. Finalmente en el módulo `slopegeometry`, la clase `AnthropicSlope` tiene los métodos `maxDepth`, `defineBoundary` y `plotSlope` y la clase `NaturalSlope` tiene los métodos `maxDepth`, `defineBoundary` y `plotSlope`.

La descripción y la forma cómo se usa cada una de las clases y sus métodos se encuentran en la documentación del PA y lo puede consultar en la dirección <https://github.com/aarizat/circpacker>.

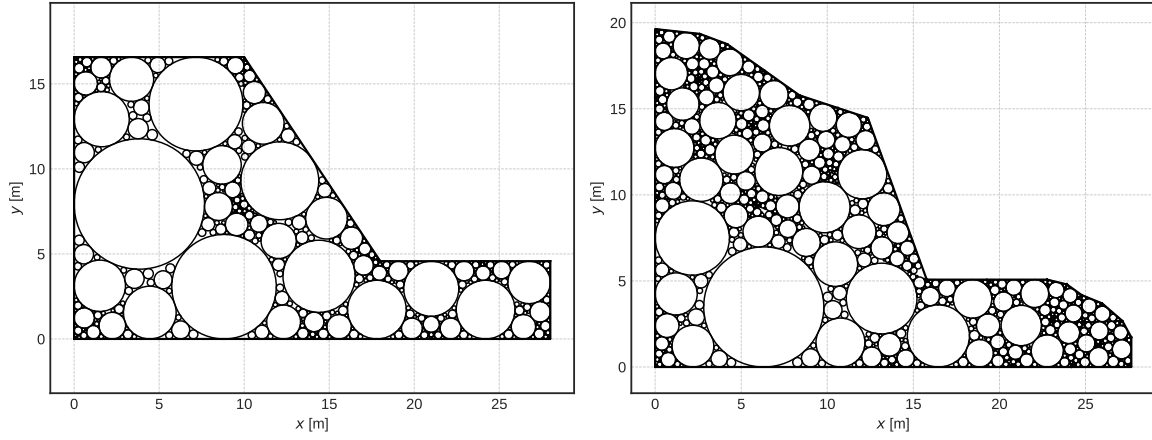
El software aplicado es libre y el código es abierto bajo la licencia BSD-2, lo cual el usuario puede entender, estudiar, modificar y mejorar el código para los fines que fue creado o para otros posibles.

4.1. Implementación del SA para generar modelos de taludes en bimsoils

Como se adelantó implícitamente en la Figura 8, un polígono irregular, generalmente cóncavo, puede representar el contorno de un talud cualquiera, bien sea de origen natural o antrópico. La Figura 9 muestra dos ejemplos: un polígono irregular de seis lados cóncavo que representa un talud simple clásico con topografía recta y un polígono de m lados cóncavo que representa una parte de una ladera natural irregular con topografía ondulada truncada por un corte.

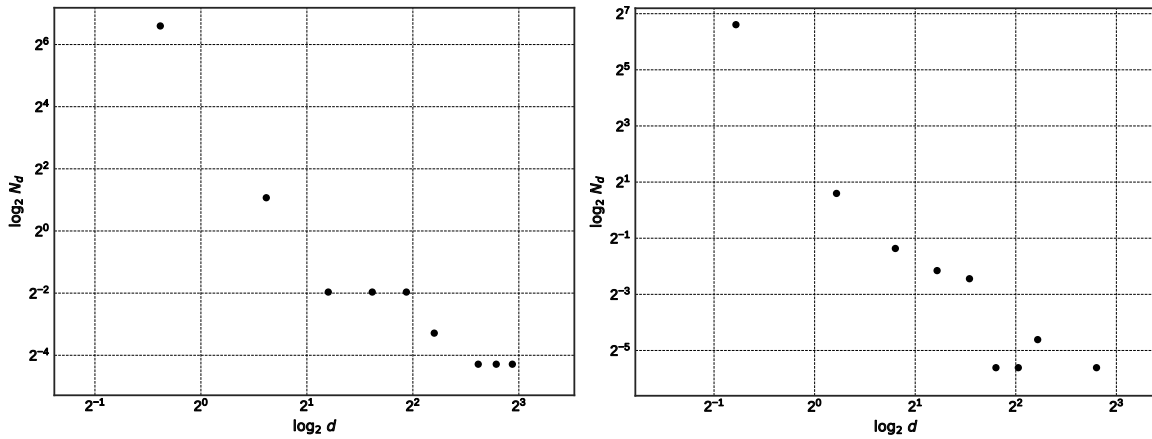
Siguiendo el procedimiento de analizar las partículas generadas, se extrae tanto los histograma de los diámetros de las partículas como las gráficas \log_2 - \log_2 de esos histogramas; de estos dos tipos de gráficas la que más información da es la gráfica \log_2 - \log_2 .

En la Figura 10 se muestra los diagramas \log_2 - \log_2 de los dos taludes modelados que se muestran en la Figura 9. En esta figura se observa que las partículas en su mayoría sí están presentes como un fractal autosimilar excepto en las partículas más grandes, lo cual se asemeja a la distribución de partículas de un *bimsoil*.



(a) Polígono irregular de seis lados cóncavo que representa un talud clásico con topografía plana. (b) Polígono de m lados cóncavo que representa una parte de una ladera irregular con topografía ondulada.

Figura 9: Llenado de polígonos en \mathbb{R}^2 con partículas circulares tangentes. Fuente: Elaboración propia.



(a) Polígono irregular de seis lados cóncavo que representa un talud clásico con topografía plana. (b) Polígono de m lados cóncavo que representa una parte de una ladera irregular con topografía ondulada.

Figura 10: Gráfica \log_2 - \log_2 de los diámetros y cantidades de las partículas circulares en los polígonos en \mathbb{R}^2 que representan taludes en *bimsoil*. Fuente: Elaboración propia.

5. DISCUSIÓN

5.1. Parámetros que controlan el comportamiento fractal

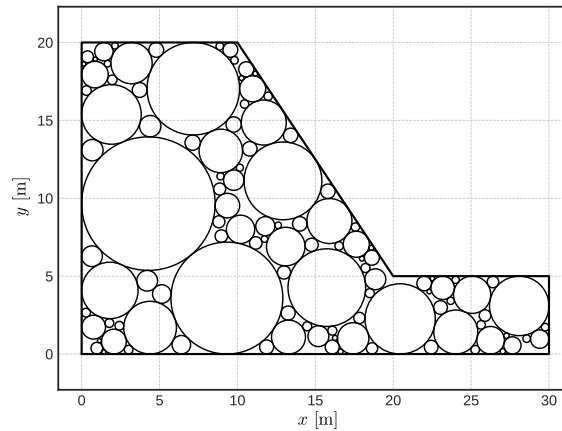
Como se ha mencionado previamente, el SA genera una malla triangular dentro de los polígonos para empaquetar los círculos; sin embargo, al observar el ejemplo de la Figura 10, la condición de autosimilaridad se pierde en los diámetros más grandes; después de realizar varias simulaciones se encontró que esta condición se repite para diferentes mallas, con lo cual se concluye que no cualquier malla triangular es capaz de generar una distribución fractal completa para todos los diámetros de los círculos empaquetados. Si bien el algoritmo es recursivo y genera círculos a varias profundidades y cumple con uno de los conceptos de fractal, este fracasa en lograr una autosimilitud completa. Amparados en estas dos condiciones parcialmente logradas, y bajo el término aquí usado, el algoritmo general *pseudofractales*.

Esta limitación (la de no lograr autosimilitud en los diámetros mayores) se puede tratar mediante la manipulación de los parámetros que controlan el enmallado para el problema particular en cuestión; es decir, el área máxima de cada triángulo (A_{\max}^{Δ} , variable `maxArea` de la clase `CircPacking` del módulo `packer.py`) y el ángulo mínimo de cualquier vértice de cada triángulo (\angle_{\min}^{Δ} , variable `minAngle` de la clase `CircPacking` del módulo `packer.py`).

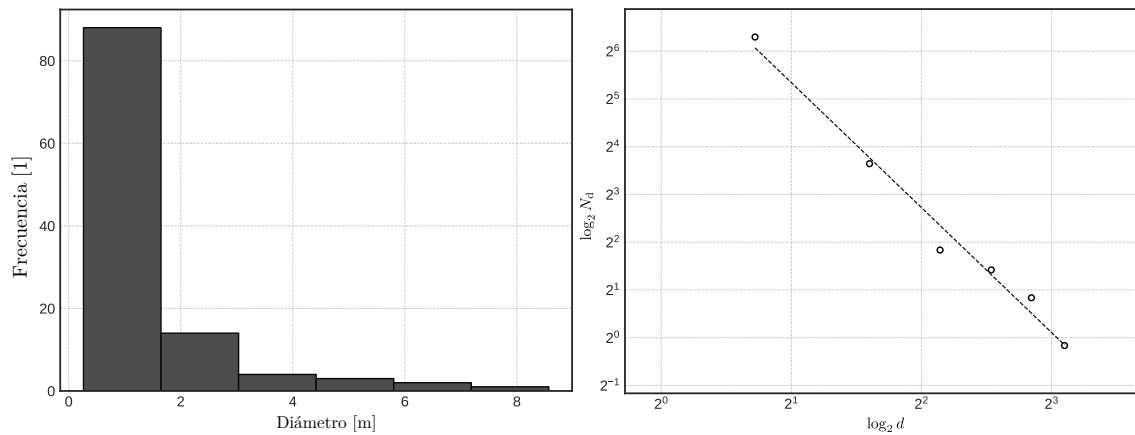
Por ejemplo, para un talud simple como el de la Figura 11(a), de altura $h = 15$ m, inclinación con relación $H : V$ de $1 : 1.5$, longitud de corona y pie de $\frac{2}{3}h$ y profundidad $\frac{1}{3}h$, se encontró que para $A_{\max}^{\Delta} = 0.35A_{\text{talud}}$ y $\angle_{\min}^{\Delta} = 20^{\circ}$, la distribución de las frecuencias de los diámetros de los círculos muestra un comportamiento exponencial (Figura 11(b)) y cumple la condición de autosimilaridad para todos los diámetros, incluyendo los más grandes, tal y como lo muestra el diagrama \log_2 - \log_2 de la Figura 11(c). Similarmente se puede encontrar tuplas de parámetros para otros taludes de tal forma que la autosimilitud se cumpla para todos los diámetros de los círculos empaquetados.

Sin embargo, la acción de *manipular los parámetros* hace que se pierda en parte el grado de automatización que se le quería dar al algoritmo. Pero aquello no hace perder la utilidad que tiene este algoritmo para generar una estructura geométrica en el plano que represente un material geológico compuesto por bloques de roca (circulares) en una matriz de suelo.

Asimismo, se encontró que mientras se probaban diferentes valores de \angle_{\min}^{Δ} , el enmallado no converge para valores más grandes que 35° ; es decir, mediante el método de la *triangulación de Delaunay* no es posible garantizar que todos los triángulos tengan longitudes similares en sus tres lados, ya que el vértice más agudo de al menos uno de los triángulos debe ser como mucho de 35° .



(a) Modelo del talud en *bimsoil*.



(b) Histogramas de los diámetros de las partículas circulares. (c) Gráfica \log_2 - \log_2 de los diámetros y cantidades de las partículas circulares

Figura 11: Modelo de *bimsoil* autosimilar para todos los diámetros, generado con el SA desarrollado. Fuente: Elaboración propia.

5.2. Verificación de la independencia de escala

Durante la investigación de (Medley, E. W. & Lindquist, E., 1995), además de la condición de autosimilaridad en las distribuciones de tamaños de los bloques de los *melanges* de San Francisco (caracterizados como *bimrocks*), también se observó que existe una condición de independencia de escala para las distribuciones de tamaños de los bloques; esta condición se observó al normalizar los diámetros por \sqrt{A} , donde A es el área analizada.

Más adelante, (Medley, E., 2001) sugirió que para diferentes problemas ingenieriles existe un valor adecuado sobre el cual se puede escalar el problema (o normalizar en el caso de analizarlo para varias escalas); este parámetro se denominó *dimensión o longitud ingenieril característica* L_c , y para el caso de taludes, se estableció que un valor adecuado sería su altura (h).

Tomando en cuenta lo anterior, se seleccionaron cinco taludes con diferentes alturas e inclinaciones, sobre los cuales se aplicó el SA desarrollado para generar modelos de *bimsoils* que cumplen la condición de autosimilaridad para todos los diámetros de los círculos (logrado tal como se explicó en la sección inmediatamente arriba). Los cinco modelos fueron normalizados por $L_c = h$, y se generaron las gráficas \log_2 – \log_2 de todos los datos, y con ella se encontró que el SA es capaz de producir modelos que cumplen la condición de independencia de escala con una dimensión o coeficiente fractal $D = 2.3$ como se aprecia en la Figura 12.

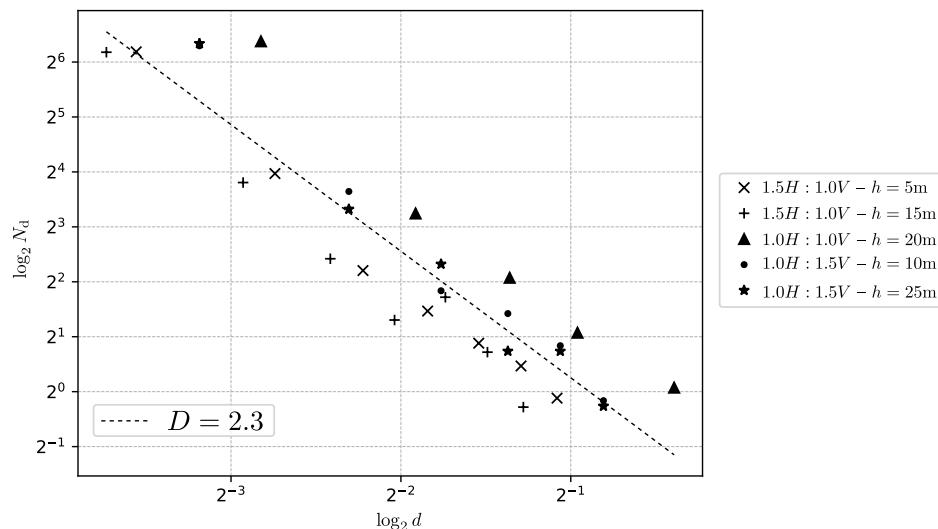


Figura 12: Demostración de la independencia de escala en los modelos de taludes en *bimsoils* Fuente: Elaboración propia.

5.3. Limitaciones sobre la tangencia de los círculos

La intención del SA es el de generar círculos tangentes con el fin de modelar *bimsoils* dentro de polígonos que inicialmente representan el contorno de taludes con aplicación posterior en análisis de estabilidad por el método bien sea de equilibrio límite o por el método de esfuerzo–deformación (*e.g.* con el auxilio del método de los elementos finitos). Inicialmente se plantea para esta labor, pero es fácilmente extensible a otras aplicaciones sobre estos materiales.

En cualquiera de las dos alternativas de análisis, la condición de la generación de la tangencia de todos los círculos no necesariamente es una ventaja. Si se opta por el análisis del talud en este material particulado por el método de equilibrio límite, la superficie de ruptura modelada no podría generarse debido a que ella no podría pasar en los puntos de contacto entre tangentes.

Similar limitación se tendría si se opta por el análisis del talud por el método de esfuerzo–deformación; de ser así, los esfuerzos se transmitirían de partícula a partícula sin tener que pasar el esfuerzos a la matriz, por lo que sería algo similar a modelar un macizo de roca.

Por tanto, para el análisis del material modelado por este generador, algunas partículas pequeñas se deben *apagar o borrar* del modelo inicial, de tal forma que no exista demasiadas tangencias entre las partículas y así poder tomar en cuenta la matriz. Apagar una partícula es omitir temporalmente la presencia de una partícula en el modelo geométrico, mientras que borrar es anular definitivamente la información de la presencia de una partícula en el modelo.

El procedimiento de apagado o borrado debe tener un criterio de escogencia. Referente a algún criterio en específico que logre mantener un valor de coeficiente fractal deseado con el apagado o borrado de partículas todavía no se ha llegado a proponer en nuestra experimentación.

Sin embargo, el criterio puede ser de elegir un umbral máximo de diámetro, donde todas las partículas que tengan diámetros menores al umbral se apagan o se borran. Otro criterio puede ser el apagado o borrado de un cierto porcentaje de partículas de cualquier diámetro escogidas al azar. Un tercer criterio puede ser una combinación de ambas pero de forma controlada y calculada, de tal forma que se logre que la relación de partículas respecto matriz sea un valor constante y deseado. Un cuarto criterio consistiría únicamente en reducir ligeramente los radios de los círculos generados, conservando el centro en su posición original.

5.4. Otros generadores de estructuras de *bimsoils*

Por otra parte, resulta interesante realizar comparaciones con otros desarrollos enfocados en la generación de modelos de *bimsoils* mediante el empaquetado de partículas circulares; no obstante, esto resulta difícil

en cuanto a que poco se ha publicado al respecto. En Napoli, M. L. *et al* (2018) se indicó que la generación de los modelos teóricos usados en sus análisis fue realizada mediante simulaciones Monte Carlo, donde los bloques son generados aleatoriamente en cuanto tamaño y posición dentro del talud siguiendo una regla estadística (no indicada) y una proporción volumétrica dada; sin embargo, no se profundiza realmente en dichas simulaciones y no se presenta de forma clara la implementación de aquello (no hay un código [una herramienta] que lo genere).

En Montoya-Araque, E. A. (2019) se empleó la librería `circlify` (Lecomte, J., 2019) desarrollada con base a Huang, W. Q. *et al* (2006), para generar los modelos empleados en sus análisis de sensibilidad; esta librería empaqueta círculos tangentes dentro de un círculo que los contiene, por lo tanto, Montoya-Araque, E. A. (2019) debió considerar un círculo que contuviera el contorno del talud para poder garantizar que el empaquetamiento abarcara todo el talud, lo cual resulta ineficiente, además de no garantizar la condición de autosimilaridad.

6. CONCLUSIONES

El Software de aplicación `circpacker` presentado en este artículo implementa el algoritmo (descrito también en este) para generar partículas circulares dentro de uno a m triángulos. Al aplicar esta regla a un enmallado triangular que llena un polígono cerrado cualquiera en \mathbb{R}^2 se logra llenar este con una serie de partículas circulares de diferentes tamaños.

Por la forma cómo se generan las partículas dentro de cada triángulo (de forma recursiva aplicando el mismo procedimiento hasta una profundidad si fuera infinita) se cumple una de las dos condiciones de un fractal. La segunda condición de fractal, que es la autosimilitud de los diámetros, no se logra de forma automática con este algoritmo e implementación. Las partículas de mayores diámetros van alejándose de la autosimilitud con este algoritmo; de este modo se ha llamado aquí a esta forma de generar arreglos empaquetados: *pseudofractal*.

Sin embargo, la implementación posibilita que apagando y borrando algunas partículas dentro de la generación automática se logre la autosimilitud, pero esto ya no es automático sino necesita de un criterio que inicialmente es solo heurístico y de prueba-error y está más relacionado con la forma de la triangulación que depende también de la forma del polígono cerrado donde se quiere general. Bajo este procedimiento, sí se logra tener partículas que son autosimilares en lo que respecta la distribución del diámetro de las partículas, pero depende de la forma del polígono cerrado.

No se ha analizado autosimilitud en lo que respecta adicionalmente *la distribución espacial de las partículas*; es decir, el algoritmo no garantiza que una distribución de diámetro de partículas (lograda de forma

parcial o total) también cumpla en su distribución espacial (*i.e.* que mantenga la distribución de diámetros a diferentes localizaciones dentro del polígono cerrado). Esto es una condición más difícil de lograr.

El algoritmo y la implementación tiene aplicabilidad para modelar la estructura del material geotécnico denominado *bimsoil*, compuesto por bloques de roca en matriz de suelo. Debido a que un *bimsoil* no rigurosamente debe ser autosimilar ni en distribución de diámetros ni espacialmente, es una herramienta útil donde se logra con cierto control y de forma automática la generación de partículas.

Este software está escrito en PYTHON3® y se lo presenta a la comunidad como un código abierto y libre.

Fue desarrollado por los autores y expuesto en el presente artículo con el fin de que su aplicación se oriente inicialmente al desarrollo del marco geométrico para el análisis de la estabilidad de taludes en *bimsoils* en dos dimensiones (donde la forma geométrica del polígono cerrado no varía sustancialmente de un patrón dado por la base plana, la pared inclinada y la corona plana). El marco geométrico posibilitará que se pueda analizar la estabilidad del talud en estos materiales mediante el método de equilibrio límite o el de esfuerzo–deformación en el plano (*i.e.* en \mathbb{R}^2).

En forma especial, la forma de aplicar esta implementación necesita de un tratamiento sobre las variables que controlan la geometría del enmallado triangular en el que se empaquetan los círculos para garantizar la condición de autosimilaridad para todos los diámetros. Por ejemplo, se confirmó la condición de independencia de escala para los *bimsoils* demostrada previamente por Medley, E. W. & Lindquist, E. (1995), al modelar taludes de diferentes inclinaciones y diferentes alturas con empaquetamientos que cumplen la condición de autosimilaridad para todos los diámetros. En específico, se encontró que esta implementación genera empaquetados con partículas circulares con un coeficiente o dimensión fractal $D = 2.3$ para polígonos que tienen una forma que representan taludes en dos dimensiones.

AGRADECIMIENTOS

A la Universidad Nacional de Colombia por los recursos de tiempo brindados para el desarrollo de esta aplicación informática y el presente artículo.

Referencias

- Bourke, P. (1992). *Spheres, equations and terminology*. <http://paulbourke.net/geometry/circlesphere/>.
- Guevara, G., Isabel, C., & Pinzón, J. (2015). Modelación numérica multiescala de Bimrocks para el análisis de estabilidad de taludes. *Master's thesis*, Pontificia Universidad Javeriana, Bogotá, Colombia.

- Huang, W. Q., Li, Y., Li, C. M., & Xu, R. C. (2006). New heuristics for packing unequal circles into a circular container. *Computers & Operations Research*, 33(8), 2125 - 2142.
- Irfan, T. Y. & Tang, K. Y. (1993). Effect of the coarse fraction on the shear strength of colluvium. *Geotechnical Engineering Office, Civil Engineering Department.*, Hong Kong.
- Kalender, A., Sonmez, H., a C. Tukuluoğlu, E. M., & Kasapoglu, K. (2014). An approach to predicting the overall strengths of unwelded bimrocks and bimsoils. *Engineering Geology*, 183, 65-79.
- Kausch-Blecken, H. H., Schmeling, V., & Tschoegl, N. W. (1970). Osculatory packing of finite areas with circles. *Nature*, 225 (5238), 1119-1122.
- Khorasani, E., Amini, M., Hossaini, M. F., & Medley, E. (2019). Evaluating the effects of the inclinations of rock blocks on the stability of bimrock slopes. *Geomechanics and Engineering*, 17(3), 279-285.
- Khorasani, E., Amini, M., Hossaini, M. F., & Medley, E. (2019). Statistical analysis of bimslope stability using physical and numerical models. *Engineering Geology*, 254, 13-24.
- Kim, C., Smell, C., & Medley, E. W. (2004). Shear strength of franciscan complex melange as calculated from back-analysis of a landslide. *In Proceedings: Fifth International Conference on Case Histories in Geotechnical Engineering*, New York.
- Lecomte, J. (2019). *circlify*. <https://pypi.org/project/circlify/>.
- Lindquist, E. (1995). The strength and deformation properties of melange. *PhD thesis*, University of California at Berkeley, Berkeley, California.
- Lucca, G. (2017). Chains of tangent circles inscribed in a triangle. *Forum Geometricorum*, 17, 41-44.
- Medley, E. (1996). The engineering caraterization of melanges and similar block-in-matrix rocks (Bimrocks). *PhD thesis*, University of California at Berkeley, Berkeley, California.
- Medley, E. (2001). Orderly characterization of chaotic franciscan melanges. *Felsbau Rock and SoilEngineering Journal for Engineering Geology, Geomechanics and Tunneling*, 19(4), 35-43.
- Medley, E. (2004). Observations on tortuous failure surfaces. *Felsbau Rock and Soil Engineering Journal for Engineering Geology, Geomechanics and Tunneling*, 22(5), 35-43.
- Medley, E. W. & Lindquist, E. (1995). The engineering significance of the scale-independence of some franciscan melanges in california, usa. *In The 35th US Symposium on Rock Mechanics (USRMS). American Rock Mechanics Association.. Symposium*.
- Medley, E. W. & Rehmann, P. F. S. (2004). Characterization of bimrocks (rock/soil mixtures) with application to slope stability problems. *In Proceedings: EUROCK 2004 & 53rd Geomechanics Colloquy*, Salzburg, Austria.

- Minuto, D. & Morandi, L. (2015). Engineering Geology for Society and Territory, volume 2: Landslide Processes, chapter Geotechnical characterization and slope stability of a relict landslide in bimsoils (blocks in matrix soils) *in downtown Genoa*, Italy, pages 1083-1088. SpringerInternational Publishing.
- Araque, M., & Andres, E. (2019). Slope stability assessment for block-in-matrix composite-materials by the tortuous surface method. *Doctoral dissertation, Universidad Nacional de Colombia-Sede Medellín*.
- Montoya-Araque, E. A. & Suarez-Burgoa, L. O. (2019). Automatic generation of tortuous failure surfaces in block-in-matrix materials for 2d slope stability assessments. *Computers and Geotechnics*, 112, 17-22.
- Napoli, M. L., Barbero, M., Ravera, E., & Scavia, C. (2018). A stochastic approach to slope stability analysis in bimrocks. *International Journal of Rock Mechanics and Mining Sciences*, 101, 41-49.
- O'Rourke, J. (1997). Computational Geometry In C, chapter 1. Polygon Triangulation, pages 1-43. *Cambridge University Press*, 2 edition.
- Rufat, D. (2019). *triangle*. <https://rufat.be/triangle/>.
- Sandig, L. (2014). *Apollonian gasket generator*. <http://lsandig.org/cgi-bin/apollon/index.cgi>.
- Tanton, J. (2005). *Encyclopedia of Mathematics*. Facts On File, Inc.