

<b>PROCESO:</b> Gestión de Investigación	<b>MANUAL</b>	<b>Código: II-MA-009</b>
<b>SUBPROCESO:</b> Gestión Editorial y Visibilidad de Publicaciones	<b>PLANTILLA PRESENTACIÓN DE ARTÍCULOS</b>	<b>Versión: 3</b>

## DESCOMPOSICIÓN DE UNA FIGURA POLIGONAL SIN HUECOS EN UN CONJUNTO DE POLÍGONOS CONVEXOS POR MEDIO DE UN ALGORITMO EXACTO Y UNO APROXIMADO

Decomposition of a polygonal figure without gaps into a set of convex polygons by means of an exact and an approximate algorithm

**Johana Milena Martínez Contreras**

Jm.martinezc1@uniandes.edu.co

Universidad de Los Andes

0009-0007-7882-8173

Colombia

Magíster en Ingeniería Industrial-Universidad de los Andes

Ingeniera Industrial-Universidad Industrial de Santander

Instructora del Departamento de Ingeniería Industrial

Grupo de Investigación en Producción y Logística (PYLO)

**David Álvarez Martínez**

d.alvarezm@uniandes.edu.co

Universidad de Los Andes

0000-0001-8411-1936

Colombia

Postdoct en Sistemas de Optimización Aplicada-Universidad Politécnico de Valencia

Ingeniero de Sistemas y Computación-Universidad Tecnológica de Pereira

Grupo de Investigación en Producción y Logística (PYLO)

**Germán Pantoja Benavides**

gf.pantoja10@uniandes.edu.co

Universidad de Los Andes

0009-0007-3177-3929

Colombia

Magíster en Ingeniería Química- Universidad de los Andes

Ingeniero Químico- Universidad de los Andes

Grupo de Investigación en Producción y Logística (PYLO)

*Recepción: 10/08/2024*

*Aceptación: 30/11/2024*

**DOI:**

“Este documento es propiedad intelectual del POLITECNICO GRANCOLOMBIANO, se prohíbe su reproducción total o parcial sin la autorización escrita de la Rectoría. TODO DOCUMENTO IMPRESO O DESCARGADO DEL SISTEMA, ES CONSIDERADO COPIA NO CONTROLADA”.

<b>PROCESO:</b> Gestión de Investigación	<b>MANUAL</b>	<b>Código: II-MA-009</b>
<b>SUBPROCESO:</b> Gestión Editorial y Visibilidad de Publicaciones	<b>PLANTILLA PRESENTACIÓN DE ARTÍCULOS</b>	<b>Versión: I</b>

## Resumen

La descomposición de una figura poligonal en un conjunto de polígonos convexos es útil en el empaquetamiento de formas irregulares y el corte de materiales. Los polígonos convexos permiten que cualquier línea entre dos puntos dentro del polígono permanezca en su interior, resaltando que la figura no es hueca. En este trabajo, proponemos un algoritmo exacto y uno aproximado mejorados para la mínima descomposición convexa de polígonos. Adaptamos un modelo exacto existente, logrando soluciones óptimas, y desarrollamos un algoritmo heurístico más rápido, aunque no siempre óptimo. También implementamos un algoritmo de mejora que une subpolígonos convexos para reducir su número. Comparamos nuestros algoritmos con otros y encontramos que la heurística combinada con la unión de convexos tuvo un desempeño superior. La solución exacta es recomendada cuando el tiempo computacional no es una limitación. Para trabajos futuros, se sugiere integrar estos algoritmos en enfoques de corte bidimensional de piezas irregulares.

## Palabras clave

Descomposición convexa, polígonos, geometría computacional, nesting.

## INTRODUCCIÓN

La descomposición de una figura poligonal sin huecos en polígonos convexos permite un análisis computacional eficiente. Esta técnica se usa en topografía (Wei, et al., 2022), en el entrenamiento de redes neuronales (Deng, et al., 2020), en el reconocimiento de patrones (Li, Zhang, Lui, & Yang, 2020) y en Strip Packing (Pantoja, Álvarez, Parreño, & Álvarez, 2022). Es fundamental determinar el número mínimo de partes convexas, disjuntas o no, para optimizar el proceso. La descomposición tiene como objetivo remover los nodos cóncavos del polígono, dividiendo la figura en formas convexas (Wei, et al., 2022).

La complejidad de la descomposición de un polígono depende del número de vértices cóncavos y de su forma. Por otro lado, si se busca la descomposición convexa de una figura libre con huecos, el problema resultante es un NP-Hard (Fernández, Cánovas, & Pelegrin, 2000). La geometría computacional estudia problemas reales usando figuras geométricas y diseña algoritmos para resolverlos. Los polígonos convexos son preferidos en gráficos computacionales por su simplicidad (Schachter, 1978).

Un polígono es convexo si todas sus diagonales y segmentos entre dos puntos están dentro de él (Lien & Amato, 2005; Falcó, 2022). Existen métodos de descomposición como la suma de Minkowski y la triangulación (Taranilla, Gagliardi, Leguizamón, & Hernández, 2007; Saracevic & Selimi, 2019); sin embargo, las metaheurísticas pueden ofrecer mejores soluciones combinando diferentes algoritmos.

Se proponen dos metodologías: un algoritmo exacto y uno aproximado. El algoritmo exacto, basado en Keil (1991), realiza descomposiciones convexas iterativas y combina triángulos contiguos. El algoritmo aproximado, basado en Fernández, et al. (2000), itera con vértices notch y vecinos no notch, evaluando y retrocediendo si no cumplen las especificaciones. El algoritmo exacto ofrece soluciones óptimas, mientras que el aproximado reduce el tiempo computacional, encontrando un conjunto cercano al óptimo. La unión de piezas convexas mejora los resultados, considerando la posibilidad de piezas disjuntas.

“Este documento es propiedad intelectual del POLITECNICO GRANCOLOMBIANO, se prohíbe su reproducción total o parcial sin la autorización escrita de la Rectoría. TODO DOCUMENTO IMPRESO O DESCARGADO DEL SISTEMA, ES CONSIDERADO COPIA NO CONTROLADA”.

<b>PROCESO:</b> Gestión de Investigación	<b>MANUAL</b>	<b>Código: II-MA-009</b>
<b>SUBPROCESO:</b> Gestión Editorial y Visibilidad de Publicaciones	<b>PLANTILLA PRESENTACIÓN DE ARTÍCULOS</b>	<b>Versión: I</b>

## MÉTODO

Se busca un algoritmo para dividir una figura bidimensional no convexa en el menor número posible de polígonos convexos y luego unirlos para reducir la cantidad total de piezas. Se ha modificado un enfoque exacto propuesto por Keil (1991), que realiza una descomposición convexa exacta dividiendo la figura en subpolígonos y triángulos para optimizar las uniones y reducir el número de piezas convexas. Además, se desarrolla una heurística para una descomposición aproximada, uniendo ángulos notch, seleccionando vértices iniciales con más vecinos y evaluando la combinación de los polígonos. Finalmente, se implementa un algoritmo de unión de piezas convexas para comparar las soluciones obtenidas por los enfoques exacto y aproximado.

Los datos de entrada son las coordenadas de los vértices, leídas en sentido horario desde el vértice 0. Se calculan las características de la figura para determinar si es convexa. El Algoritmo 1 es el algoritmo exacto modificado, y el Algoritmo 2 calcula la visibilidad entre los puntos. La visibilidad implica conectar dos puntos sin obstáculos dentro de la figura. Juntos, evalúan colinealidad y validez de los subpolígonos, organizándolos por cantidad de vértices. Luego, encuentran los triángulos base de cada subpolígono y generan la descomposición convexa mínima con una complejidad de  $O(n^2)$ .

El Algoritmo 3, basado en Fernández (2000), intenta unir notches consecutivos para formar figuras convexas, buscando una descomposición convexa más rápida que el método exacto. Aunque no siempre es óptimo, ofrece soluciones cercanas al óptimo en menos tiempo. Ambos algoritmos son útiles: el exacto ofrece soluciones óptimas y el aproximado permite obtener soluciones cercanas al óptimo en un tiempo razonable. Por otro lado, se propone un Algoritmo 4, que hace la unión de las piezas divididas convexas con el fin de disminuir aún más el conjunto de figuras convexas, si el problema a tratar con estos algoritmos así lo permite.

## RESULTADOS

Los algoritmos se programaron en Python 3.9.12 y se ejecutaron en un procesador Intel Core i7-4702MQ con 16 GB de RAM. Se probaron en 250 instancias divididas en cinco grupos según el número de vértices, utilizando las mismas instancias que Fernández (2000), cada grupo tiene una media, un máximo y un mínimo de notches.

Se compararon el algoritmo exacto (Keil, 1991) sin y con unión de piezas (Algoritmo 1 y Algoritmo 4). La unión no redujo significativamente el número de piezas ni aumentó el tiempo de ejecución. Adicionalmente, se compararon los algoritmos, exactos (Algoritmo 1) y aproximado (Algoritmo 3), como se muestra en la Tabla 1. El GAP entre el Algoritmo 3 y el Algoritmo 1 varía entre 11.86% y 22.02% para figuras con 50 y 150 vértices, respectivamente. Además, el GAP entre el Algoritmo 3 y el de Fernández (2000) es del 2.64% en la función objetivo, con una mejora del 139.6% en tiempos computacionales. Aunque el Algoritmo 1 tiene mejores resultados en la función objetivo, el Algoritmo 3 reduce significativamente los tiempos de ejecución.

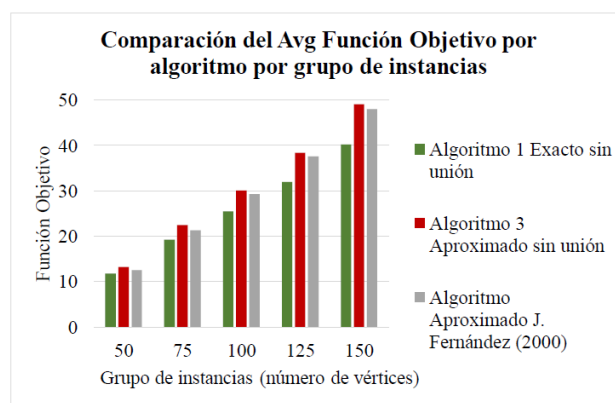
“Este documento es propiedad intelectual del POLITECNICO GRANCOLOMBIANO, se prohíbe su reproducción total o parcial sin la autorización escrita de la Rectoría. TODO DOCUMENTO IMPRESO O DESCARGADO DEL SISTEMA, ES CONSIDERADO COPIA NO CONTROLADA”.

<b>PROCESO:</b> Gestión de Investigación	<b>MANUAL</b>	<b>Código:</b> II-MA-009
<b>SUBPROCESO:</b> Gestión Editorial y Visibilidad de Publicaciones	<b>PLANTILLA PRESENTACIÓN DE ARTÍCULOS</b>	<b>Versión:</b> I

N° Vértices	Algoritmo 1 Exacto sin unión		Algoritmo 3 Aproximado sin unión				Algoritmo Aproximado J. Fernández (2000)		
	Avg Fun. Obj	Avg t [s]	Avg Fun. Obj	Avg t [s]	Dif % Avg t [s]	GAP Avg Fun. Obj	Avg Fun. Obj	Avg t [s]	GAP Avg Fun. Obj
50	11.80	0.56	13.20	0.17	229.92%	11.86%	12.52	0.48	6.10%
75	19.22	1.63	22.46	0.53	207.66%	16.86%	21.30	1.24	10.82%
100	25.5	3.53	30.12	0.97	265.11%	18.12%	29.30	2.41	14.90%
125	32.00	5.76	38.40	1.64	250.55%	20.00%	37.62	4.16	17.56%
150	40.24	5.01	49.10	3.10	61.55%	22.02%	48.04	7.43	19.38%

*Tabla 1  
Comparación entre algoritmos.*

La Ilustración 1 muestra que el Algoritmo 1 siempre obtiene los mejores valores en la función objetivo, mientras que el Algoritmo 3 y el de Fernández (2000) tienen valores similares, con el Algoritmo 3 logrando los menores tiempos de ejecución. Es importante mencionar que en todos los casos el Algoritmo 3 tuvo los menores tiempos de ejecución.



*Ilustración 1  
Comparación del Avg Función Objetivo por algoritmo por grupo de instancias.*

La Tabla 2 compara el Algoritmo 3 sin unión con los resultados al unir piezas (Algoritmo 3 y 4). La unión es más efectiva en figuras con 150 vértices y menos en las de 125.

<b>PROCESO:</b> Gestión de Investigación	<b>MANUAL</b>	<b>Código:</b> II-MA-009
<b>SUBPROCESO:</b> Gestión Editorial y Visibilidad de Publicaciones	<b>PLANTILLA PRESENTACIÓN DE ARTÍCULOS</b>	<b>Versión:</b> I

N° Vértices	Algoritmo 3 Aproximado sin unión		Algoritmo 3 Aproximado con unión		
	Avg Fun. Obj	Avg t [s]	Avg Fun. Obj	Avg t [s]	% Dif Fun. Obj
50	13.20	0.17	12.90	0.16	2.33%
75	22.46	0.53	21.96	0.51	2.28%
100	30.12	0.97	29.02	1.02	3.79%
125	38.40	1.64	38.24	1.62	0.42%
150	49.10	3.10	46.84	3.06	4.82%

Tabla 2  
Comparación algoritmo aproximado sin y con unión.

## DISCUSIÓN Y CONCLUSIÓN

El algoritmo modificado de Keil (1991) alcanza soluciones óptimas como algoritmo exacto, mientras que los enfoques aproximados no garantizan soluciones óptimas ni piezas completamente convexas. El Algoritmo 3 propuesto mejora la precisión y confiabilidad, y la unión de piezas es útil en aplicaciones como empaquetamiento y robótica. Dos resultados clave son: una disminución promedio de 2 piezas con el Algoritmo 1 y de 25 piezas con el Algoritmo 3, indicando un impacto significativo en la reducción de piezas. Además, los tiempos de ejecución son similares con y sin la unión de piezas, sugiriendo que la unión no añade tiempo significativo al algoritmo.

Futuras investigaciones podrían adaptar el algoritmo para figuras huecas y tridimensionales, que son NP-Hard y requerirían heurísticas o metaheurísticas. También se sugiere explorar la unión de más de dos piezas y desarrollar estrategias para mejorar este proceso.

## REFERENCIAS BIBLIOGRÁFICAS

- Deng, B., Genova, K., Hinton, G., Yazdani, S., Tagliasacchi, A., & Bouaziz, S. (2020). CvxNet: Learnable Convex Decomposition. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Falcó, J. (2022). *Experimental Mathematics: Manipulative activities to visualize concepts (Vol. 16)*. Ediciones SM Spain.
- Fernández, J., Cánovas, L., & Pelegrin, B. (2000). Algorithms for the decomposition of polygon into convex polygons. *European Journal of Operational Research*.
- Keil, M. (1991). Decomposing a polygon into simpler components. *Journal of Algorithms*, (págs. 71-79).
- Li, Z., Zhang, Z., Lui, H., & Yang, L. (2020). A new path planning method based on concave polygon convex decomposition and artificial bee colony algorithm. *International Journal of Advanced Robotic Systems*.
- Lien, J., & Amato, N. (2005). Approximate convex decomposition of polygons. *European Journal of Operational Research*.
- Pantoja, G., Álvarez, R., Parreño, F., & Álvarez, D. (2022). Nueva meta-heurística para solucionar el Strip Packing Problem con piezas irregulares (SPPI). *IV Congreso Colombiano de Investigación Operativa ASOCIO 2022 - IISE REGIÓN 16*.
- Saracevic, M., & Selimi, A. (2019). Convex polygon triangulation based on planted trivalent binary tree and ballot problem. *Turkish Journal of Electrical Engineering and Computer Sciences*.

“Este documento es propiedad intelectual del POLITECNICO GRANCOLOMBIANO, se prohíbe su reproducción total o parcial sin la autorización escrita de la Rectoría. TODO DOCUMENTO IMPRESO O DESCARGADO DEL SISTEMA, ES CONSIDERADO COPIA NO CONTROLADA”.

<b>PROCESO:</b> Gestión de Investigación	<b>MANUAL</b>	<b>Código: II-MA-009</b>
<b>SUBPROCESO:</b> Gestión Editorial y Visibilidad de Publicaciones	<b>PLANTILLA PRESENTACIÓN DE ARTÍCULOS</b>	<b>Versión: I</b>

Schachter, B. (1978). Decomposition of polygons into convex sets. *IEEE Transactions on Computers*.

Taranilla, M., Gagliardi, E., Leguizamón, M., & Hernández, G. (2007). Descomposición de Minkowski usando Algoritmos Genéticos. *In IX Workshop de Investigadores en Ciencias de la Computación*.

Wei, Z., Ding, S., Cheng, L., Xu, W., Wang, Y., & Zhang, L. (2022). Linear building pattern recognition in topographical maps combining convex polygon decomposition. *Geocarto International*, 1-25.