

## *Una operación entre polígonos: Sumas de Minkowski*

María Teresa Taranilla <sup>(1)</sup>

Gregorio Hernández Peñalver

Gustavo Kavka <sup>(1)</sup>

Edilma Olinda Gagliardi <sup>(2)</sup>

<sup>(1)</sup> Departamento de Informática

Departamento de Matemática Aplicada

<sup>(2)</sup> LIDIC<sup>♦</sup>

Facultad de Ciencias Físico, Matemáticas y Naturales

Facultad de Informática

Universidad Nacional de San Luis, Argentina

Universidad Politécnica de Madrid, España

{ tarani, gkavka, oli }@unsl.edu.ar

gregorio@fi.upm.es

Fax: 54-2652-430224

Fax: 34-91-3367426

### Resumen:

Dados dos conjuntos  $P$  y  $Q \subset \mathbb{R}^2$ , la suma de Minkowski de  $P$  y  $Q$ , denotada por  $P \oplus Q$  se define como  $P \oplus Q = \{ p + q : p \in P, q \in Q \}$ .

Las sumas de Minkowski son utilizadas en un amplio rango de aplicaciones, tales como planificación de movimientos de robots, procesamiento de imágenes, sistemas de información geográfica, marcado y corte de moldes, entre otras.

En este trabajo se presentan las *Sumas de Minkowski*, mostrando aspectos teóricos, sus posibles aplicaciones, se describen los algoritmos que las calculan y finalmente, se introduce una herramienta para el cálculo de sumas de Minkowski entre polígonos.

**Palabras claves:** Sumas de Minkowski, Geometría Computacional.

---

<sup>♦</sup> Laboratorio de Investigación y Desarrollo en Inteligencia Computacional. Director: Dr. Raúl H. Gallard.  
Este artículo es parcialmente suvencionado por el Proyecto AI2002-1010-2.43 Geometría Computacional, de la Universidad Politécnica de Madrid, España.

## 1. Introducción

La Geometría Computacional es una disciplina que brinda un marco teórico y formal para el diseño de estructuras y análisis de algoritmos requeridos para dar soluciones a problemas que surgen en las más diversas áreas de la Informática. Actualmente ha cobrado un gran interés debido a las numerosas aplicaciones que tiene en distintas áreas de investigación [Abe00], [BKOS97] [Tou85], [Tou92].

La Geometría Computacional se ocupa de resolver problemas geométricos de modo constructivo. El volumen de datos que se maneja en muchas aplicaciones suele ser muy elevado, por lo que es necesario buscar una solución basada en dos parámetros que midan la complejidad de un algoritmo: el espacio y el tiempo[AHU74].

En este trabajo se hace hincapié en un tema particular de la Geometría que son las *sumas de Minkowski*. Las sumas de Minkowski son utilizadas en un amplio rango de aplicaciones, tales como planificación de movimientos de robots [Lat91], procesamiento de imágenes [Ser82], [Ser88], sistemas de información geográfica [HCC98], marcado y corte de moldes[Li94], entre otras.

Uno de los objetivos de nuestro trabajo, fue introducirnos en el estudio e investigación de las sumas de Minkowski, su contexto teórico, aplicaciones más destacadas y realizar un estudio que muestre el estado del arte del tema presentando los aspectos teóricos y prácticos más relevantes [GTBH02] [BS01], [AFH01].

Otro de los objetivos planteados, fue desarrollar una herramienta que implemente la suma de Minkowski entre distintos tipos de polígonos, tanto entre polígonos convexos y no convexos, que pueda ser utilizada como una herramienta de trabajo, de simulación y de apoyo en la enseñanza de la Geometría Computacional [KT02].

En el presente artículo, introducimos el concepto de sumas de Minkowski y describimos algunas de sus aplicaciones más relevantes. Luego, presentamos los algoritmos que permiten realizar el cálculo de la suma de Minkowski entre polígonos convexos y no convexos. Finalmente, introducimos la herramienta desarrollada para el cálculo y la visualización de las Sumas de Minkowski.

## 2. Sumas de Minkowski y sus aplicaciones

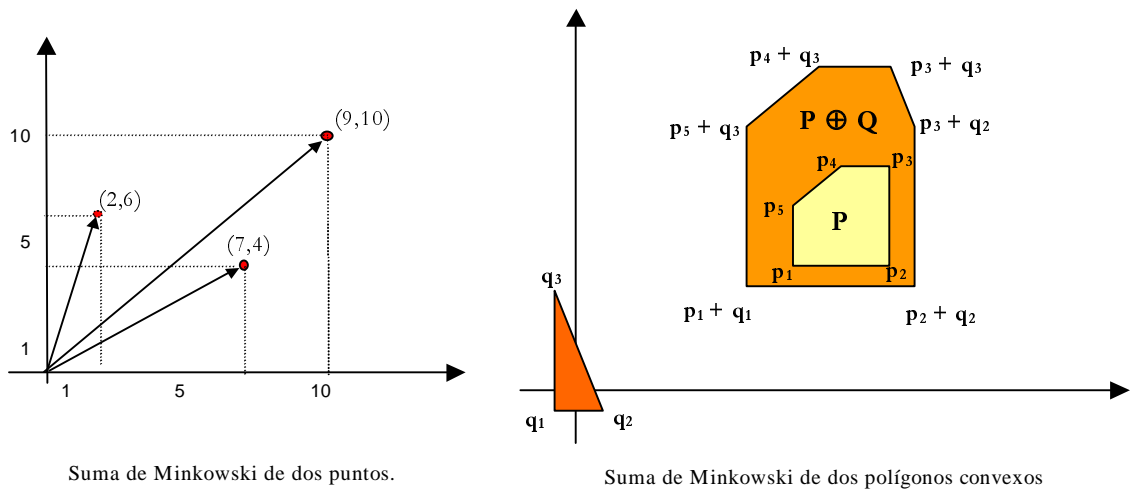
Dados dos conjuntos  $P$  y  $Q \subset \mathbf{R}^2$ , la suma de Minkowski de  $P$  y  $Q$ , denotada por  $P \oplus Q$  se define como  $P \oplus Q = \{ p + q : p \in P, q \in Q \}$  donde  $p + q$  es un vector que representa la suma de los vectores  $p$  y  $q$ . Es decir que dados los puntos  $p = ( p_x, p_y )$  y  $q = ( q_x, q_y )$ , tenemos que  $p + q = ( p_x + q_x, p_y + q_y )$ .

En las figura 2.1 podemos observar a la izquierda la suma de Minkowski entre dos puntos en el plano y en la figura de la derecha la suma de Minkowski entre dos polígonos convexos.

Las áreas de aplicación de las sumas de Minkowski se han remitido principalmente a la robótica, específicamente en planificación de movimientos [Lat91], [BKOS97].

El problema de la planificación de movimiento de robots consiste en encontrar un movimiento continuo para transportar un objeto desde un punto de partida a un punto de llegada, tal que el objeto no colisione con ningún obstáculo en su camino.

El problema general de la planificación de movimiento de robots es bastante difícil por lo que se harán simplificaciones.



**Figura 2.1**

La simplificación más drástica es considerar el plano en dos dimensiones. El entorno del robot será una región plana con obstáculos que son un conjunto de polígonos disjuntos, incluso el robot  $R$  será poligonal. Asumiremos también que el entorno es estático, no habrá gente caminando en el camino del robot. Nos restringiremos a tratar con robots que solamente se trasladan.

Sea  $R$  un robot que se mueve a través de un espacio bidimensional, o *espacio de trabajo*, que consiste de un conjunto  $S = \{P, P_2, \dots, P_t\}$  de obstáculos poligonales. Asumimos también que  $R$  es un polígono simple. Debemos saber dónde está ubicado el robot en el espacio de trabajo, para ello consideraremos al robot en términos de un *punto de referencia* ubicado en el interior del robot. De este modo, vamos a expresar la *ubicación* de un robot  $R$  simplemente dando las coordenadas del punto de referencia cuando el robot está en una posición determinada. Así  $R(x,y)$  significa que el robot está localizado con su punto de referencia en  $(x,y)$ .

El *espacio de configuración* de un robot  $R$  de traslación en el plano es el plano Euclidiano bidimensional, y por lo tanto idéntico espacio de trabajo. Es útil distinguir las dos nociones: el espacio de trabajo es el espacio donde el robot realmente se mueve y el espacio de configuración es el espacio parámetro del robot. Un robot poligonal en el espacio de trabajo se representa por un punto en el espacio de configuración, y cualquier punto en el espacio de configuración corresponde a alguna ubicación del robot en el espacio de trabajo.

Asumiendo que el robot con el cual estamos trabajando es un polígono convexo y que los obstáculos también son polígonos convexos, definimos el *espacio de configuración del obstáculo o C-obstáculo* de un obstáculo  $P$  y un robot  $R$  como el conjunto de puntos en el espacio de configuración tales que si ubicamos a  $R$  en ese punto intersecta a  $P$ . Lo denotaremos como  $CP$ , entonces tenemos que  $CP = \{ (x,y) / R(x,y) \cap P \neq \emptyset \}$ .

Podemos visualizar la forma del  $CP$  colocando a  $R$  alrededor del borde de  $P$ . La curva que se forma pasando por los puntos de referencia de  $R$  muestran el borde de  $CP$ , como se puede observar en la figura 2.2.

A continuación, mostraremos la utilidad de la suma de Minkowski para describir el espacio de configuración del obstáculo  $P$ , pero antes de expresar a  $CP$  como sumas de Minkowski vamos a definir alguna notación que necesitaremos.

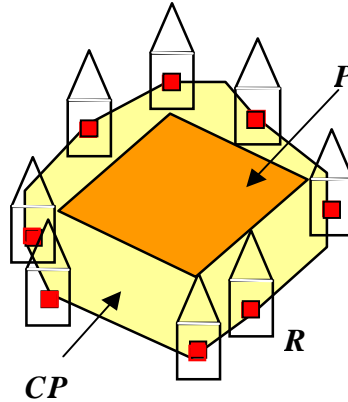


Figura 2.2 : Espacio de configuración de obstáculo del obstáculo P

Dado un punto  $p$  con coordenadas  $(x,y)$  definimos  $-p$  como el par  $(-x,-y)$  y para un conjunto  $S \subset \mathbf{R}^2$ , definimos  $-S = \{ -p / p \in S \}$ , es decir que obtenemos  $-S$  reflejando  $S$  respecto del origen.

Para  $R(0,0)$ , el robot  $R$  ubicado con su punto de referencia en el origen,  $-R(0,0)$  es el robot  $R$  reflejado en el origen.

**Teorema:** Sea  $R$  un robot en el plano, que sólo se traslada, y sea  $P$  un obstáculo poligonal. El espacio de configuración del obstáculo o  $C$ -obstáculo de  $P$  es  $P \oplus (-R(0,0))$ . [BKOS97]

Demostración:

Lo que debemos probar es que  $R(x,y)$  intersecta a  $P$  si y sólo si  $(x,y) \in P \oplus (-R(0,0))$ .

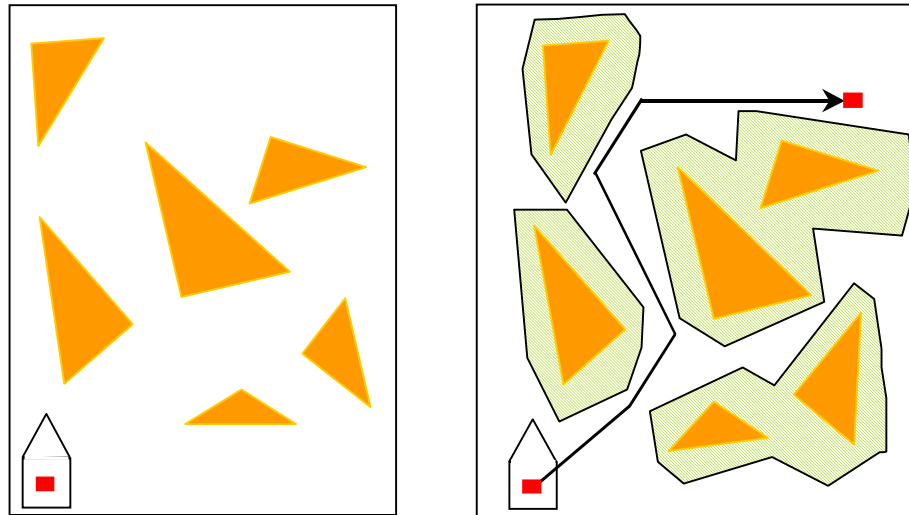
Primero supongamos que  $R(x,y)$  intersecta a  $P$  y sea  $q = (q_x, q_y)$  un punto de la intersección. Esto es porque  $q \in R(x,y)$ , así que  $(q_x - x, q_y - y) \in R(0,0)$  o equivalentemente,  $(-q_x + x, -q_y + y) \in -R(0,0)$ . Ya que  $q \in P$ , esto implica que  $(x,y) \in P \oplus (-R(0,0))$ .

Inversamente, sea  $(x,y) \in P \oplus (-R(0,0))$ . Entonces hay puntos  $r = (r_x, r_y) \in R(0,0)$  y  $p = (p_x, p_y) \in P$  tales que  $(x,y) = (p_x - r_x, p_y - r_y)$ , o en otras palabras, tales que  $p_x = r_x + x$  y  $p_y = r_y + y$ , lo cual implica que  $R(x,y)$  intersecta a  $P$ .  $\square$

Así que, el espacio de configuración de obstáculos o  $C$ -obstáculos para un robot  $R$  planar que sólo se traslada, son las sumas de Minkowski entre de los obstáculos y  $(-R(0,0))$ .

Un camino libre de colisiones traza una curva en el espacio libre. La figura 2.3 ilustra esta idea para un robot plano. En el espacio de configuración ubicado en la parte derecha de la figura, se muestra un camino sin obstáculos para un robot puntual, siendo la parte sombreada el espacio prohibido. El área no sombreada comprendida entre la sombreada es el espacio libre.

Entonces si tenemos un robot de traslación  $R$  que se mueve en un ambiente donde los obstáculos son un conjunto de polígonos disjuntos, el espacio de obstáculos es el conjunto de puntos en los cuales el robot colisiona con un obstáculo  $P$  y podemos expresar el espacio de obstáculos que corresponde a un obstáculo  $P$  como la suma de Minkowski del obstáculo con el robot  $R$ . Luego, el espacio prohibido para el robot  $R$ , puede ser descrito como la unión de las sumas de Minkowski de cada uno de los obstáculos  $P$  con el robot  $R$ .



**Figura 2.3**

*Robot y obstáculos: El robot y el conjunto de obstáculos en la figura izquierda.*

*El espacio de configuración de obstáculos y un camino libre de obstáculos para el robot en la parte derecha.*

Otra área de aplicación es en la de procesamiento de imágenes. La dilatación de un conjunto  $A$  con un elemento estructurado  $B$  es la suma de Minkowski de  $A$  y  $B$ . El elemento estructurado  $B$  es usualmente una figura simple en el plano como un cuadrado o un disco [Ser82],[Ser88].

También en los Sistemas de Información Geográfica (GIS), el término *buffer* es comúnmente usado para denotar la suma de Minkowski de un conjunto de objetos geométricos dado con un disco [HCC98].

La suma de Minkowski es herramienta poderosa de preprocesamiento usada para resolver problemas de intersección e inclusión de polígonos. Una consulta de solapamiento y/o inclusión de polígonos puede ser resuelta usando esta operación, con una complejidad sublineal. Las sumas de Minkowski son de gran utilidad en problemas de empaquetamiento [Li94].

### 3. Algoritmos para el cálculo de sumas de Minkowski

#### 3.1 Suma de Minkowski de dos polígonos convexos

Dados dos polígonos convexos  $P$  y  $Q$  con  $n$  y  $m$  vértices respectivamente, veamos cómo calcular  $P \oplus Q$ .

Veamos un primer algoritmo para calcular la suma de Minkowski de dos polígonos convexos. El algoritmo que llamaremos *algoritmo de fuerza bruta*, se basa en la siguiente idea: teniendo dos polígonos convexos  $P$  y  $Q$ , para todos los vértices de  $P$  y  $Q$  armamos pares de vértices, uno de  $P$  y otro de  $Q$ , y calculamos la suma de sus coordenadas, obteniendo  $nm$  puntos candidatos para los vértices de  $P \oplus Q$ . Luego, calculamos el cierre convexo de esos puntos candidatos y obtenemos un nuevo polígono que es la suma de Minkowski de  $P$  y  $Q$ .

**Algoritmo (de fuerza bruta):**

**Entrada:** Un polígono convexo  $P$  con  $n$  vértices  $v_1, \dots, v_n$ , y un polígono convexo  $Q$  con  $m$  vértices  $w_1, \dots, w_m$ . Se asume que la lista de vértices estará ordenada siguiendo el sentido contrario al de las agujas del reloj, siendo  $v_1$  y  $w_1$  los vértices con menor coordenada  $y$ .

**Salida:** La suma de Minkowski de  $P \oplus Q$ .

1.  $i \leftarrow 1; j \leftarrow 1; k \leftarrow 1;$
2. **Para**  $i$  desde 1 hasta  $n$  **hacer**
3.     **Para**  $j$  desde 1 hasta  $m$  **hacer**
4.          $s_k \leftarrow (v_i + w_j)$
5.          $j \leftarrow (j + 1)$
6.          $k \leftarrow (k + 1)$
7.     **Fin para**
8.      $i \leftarrow (i + 1)$
9. **Fin para**
10.  $P \oplus Q \leftarrow \text{Cierre\_Convexo}(s_1, \dots, s_k)$

Esta primera idea para un algoritmo es muy sencilla pero bastante ineficiente cuando los polígonos tienen muchos vértices ya que la complejidad de ejecución de este algoritmo es de orden cuadrático.

Es posible mejorar el costo del algoritmo anterior y para ello nos basaremos en la siguiente observación acerca de los puntos extremos de la suma de Minkowski de dos polígonos.

**Observación:** Para una dirección dada, un punto extremo en  $P \oplus Q$  es la suma de los puntos extremos de  $P$  y de  $Q$  en la misma dirección.

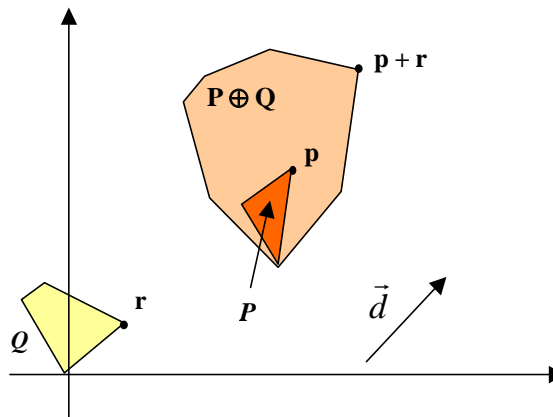


Figura 3.1: Puntos extremos de  $P \oplus Q$

Haciendo uso de esta observación es posible demostrar que la suma de Minkowski de dos polígonos convexos tiene complejidad lineal.

**Teorema:** Sean  $P$  y  $Q$  dos polígonos convexos con  $n$  y  $m$  aristas respectivamente. La suma de Minkowski  $P \oplus Q$  es un polígono convexo con  $n + m$  aristas.

**Demostración:** La convexidad de la suma de Minkowski de dos conjuntos convexos surge directamente de la definición.

Para ver que la complejidad de las Sumas de Minkowski es lineal, consideremos una arista  $e$  de  $P \oplus Q$ . Esta arista es extrema en la dirección de su salida normal. Así que debe ser generada por puntos de  $P$  y  $Q$  que son extremos en la misma dirección. Más aún, al menos uno de  $P$  y  $Q$  debe tener una arista que sea extrema en esa dirección. Nombramos  $e$  a esa arista. De esta manera cada arista es nombrada alguna vez, así que el número total de aristas es a lo más  $n+m$  ( si  $P$  y  $Q$  no tienen aristas paralelas entonces el número de aristas de la suma de Minkowski es exactamente  $m+n$  ).  $\square$

Entonces, podemos pensar en mejorar el algoritmo anteriormente planteado, mirando sólo los pares de vértices que son extremos en una misma dirección.

En un polígono convexo, si cambiamos la dirección  $\vec{d}$  siguiendo el sentido contrario a las agujas del reloj, obtenemos una secuencia de puntos extremos que contienen los vértices del polígono ordenados exactamente como están ubicados si recorremos el borde del polígono en el sentido contrario a las agujas del reloj.

El siguiente algoritmo, que llamaremos *algoritmo mejorado* recorre las direcciones siguiendo el sentido contrario a las agujas del reloj y usa la observación anterior para recorrer ambos polígonos y encontrar sus puntos extremos.

En este algoritmo vamos a agregar la notación  $\text{ángulo}(pq)$  para denotar el ángulo que forma el vector  $\overrightarrow{pq}$  con el eje positivo de abscisas.

**Algoritmo (mejorado):**

**Entrada:** Un polígono convexo  $P$  con vértices  $v_1, \dots, v_n$ , y un polígono convexo  $Q$  con vértices  $w_1, \dots, w_m$ . Se asume que la lista de vértices estará ordenada siguiendo el sentido contrario al de las agujas del reloj, siendo  $v_1$  y  $w_1$  los vértices con menor coordenada  $y$ .

**Salida:** La suma de Minkowski de  $P \oplus Q$

1.  $i \leftarrow 1 ; j \leftarrow 1$
2.  $v_{n+1} \leftarrow v_1 ; w_{m+1} \leftarrow w_1$
3. **Repetir**
4.     Agregar  $v_i + w_j$  como un vértice en  $P \oplus Q$
5.     **Si**  $\text{ángulo}(v_i v_{i+1}) < \text{ángulo}(w_j w_{j+1})$
6.         **entonces**  $i \leftarrow (i + 1)$
7.     **sino Si**  $\text{ángulo}(v_i v_{i+1}) > \text{ángulo}(w_j w_{j+1})$
8.         **entonces**  $j \leftarrow (j + 1)$
9.     **sino**  $i \leftarrow (i + 1)$
10.          $j \leftarrow (j + 1)$
11.     **Si**  $i > n+1$  **entonces**  $i \leftarrow n+1$
12.     **Si**  $j > m+1$  **entonces**  $j \leftarrow m+1$
13. **hasta que**  $i = n + 1$  y  $j = m + 1$

Este algoritmo es sencillo y fácil de implementar. Se ejecuta en un tiempo lineal  $O(m+n)$  ya que en cada ejecución de la sentencia de repetición ocurre que o  $i$  o  $j$  son incrementados sólo hasta alcanzar los valores  $n+1$  y  $m+1$ . Cualquier vértice de la suma de Minkowski  $P \oplus Q$  es la suma de dos vértices originales que son extremos en una dirección común en  $P$  y  $Q$ . Ya que los polígonos son convexos el testeo del ángulo asegura que esos pares extremos son encontrados.

En la implementación de este algoritmo, usamos una variación especial del método de rotación de calibres [Tou83] para encontrar los puntos extremos. El testeo de los ángulos se lleva a cabo usando el área signada de un triángulo.<sup>1</sup>

### 3.2 Suma de Minkowski de un polígono convexo y uno no convexo

Consideremos la suma de Minkowski de un polígono no convexo  $P$  y un polígono convexo  $R$  con  $n$  y  $m$  vértices respectivamente.

En este caso, para calcular la suma de Minkowski, primero debemos reducir el caso a suma de polígonos convexos. Para ello vamos a triangular el polígono no convexo  $P$ , así obtenemos  $n-2$  triángulos. Luego, calculamos la suma de Minkowski de cada triángulo  $t_i$  obtenido con el polígono convexo  $R$ , usando el algoritmo mejorado que calcula la suma de dos polígonos convexos con un orden lineal. La suma de Minkowski de  $P$  y  $R$  será igual a la unión de esas sumas,

$$P \oplus R = \bigcup_{i=1}^{n-2} t_i \oplus R$$

#### **Algoritmo (Suma de Minkowski de un polígono convexo con uno no convexo)**

**Entrada:** Un polígono no convexo  $P$  con  $n$  vértices  $v_1, \dots, v_n$ , y un polígono convexo  $R$  con  $m$  vértices  $w_1, \dots, w_m$ .

**Salida:** La suma de Minkowski de  $P \oplus R$

1. Triangular el polígono no convexo  $P$ , obteniendo  $t_{n-2}$  triángulos.
2. **Para**  $i$  **desde** 1 **hasta**  $n-2$
3.      $Q_i \leftarrow$  calcular  $t_i \oplus R$
4. **Fin para**
5.  $P \oplus Q \leftarrow$  calcular  $\bigcup_{i=1}^{n-2} Q_i$

Analicemos la complejidad de este algoritmo: triangular el polígono  $P$  en el paso 1 puede hacerse en orden lineal, luego en el paso 3 el cálculo de la suma de Minkowski de cada triángulo  $t_i$  con  $R$  lleva un tiempo de  $O(m)$ , y como este cálculo se repite  $n$  veces, la complejidad de este paso es de  $O(n m)$ . Por último debemos analizar cuál es el costo de calcular la  $\bigcup_{i=1}^{n-2} Q_i$ . La unión de dos polígonos convexos de  $r$  y  $s$  lados toma un tiempo de  $O(r+s)$ . Como cada polígono  $Q_i$  tiene  $m + 3$  lados, el cálculo de  $\bigcup_{i=1}^{n-2} Q_i$  que es igual a  $P \oplus Q$ , se puede llevar a cabo en un tiempo de  $O(n m)$ .

Esto implica que la suma de Minkowski de un polígono no convexo  $P$  y un polígono convexo  $R$  con  $n$  y  $m$  vértices  $P \oplus R$  tiene una complejidad de  $O(n m)$ . Este límite de la complejidad se alcanza en el peor de los casos.

---

<sup>1</sup> Se define área signada de un triángulo como  $\Delta p_1 p_2 p_3$  como  $\Delta(p_1, p_2, p_3) = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix}$



En la figura 3.2 podemos observar la suma de Minkowski de un polígono convexo  $R$  con  $m$  vértices y un polígono no convexo  $P$  de tipo *peineta* con  $n$  vértices.

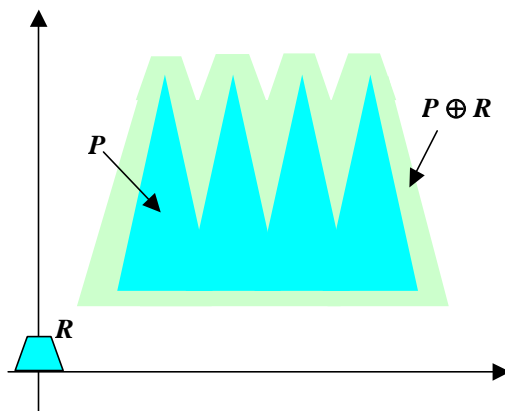


Figura 3.2: Suma de Minkowski de un polígono convexo con uno no convexo

### 3.3 Suma de Minkowski de dos polígonos no convexos

En el caso de tener de dos polígonos no convexos  $P$  y  $R$ , calcular la suma de Minkowski de ambos no es difícil; se deben triangular ambos polígonos, calcular la suma de Minkowski de cada par de triángulos, usando el algoritmo mejorado para calcular la suma de Minkowski de dos polígonos convexos y calcular la unión de los polígonos obtenidos en cada una de las sumas.

#### Algoritmo (Suma de Minkowski entre dos polígonos no convexos)

**Entrada:** Un polígono no convexo  $P$  con  $n$  vértices  $v_1, \dots, v_n$ , y un polígono no convexo  $R$  con  $m$  vértices  $w_1, \dots, w_m$ .

**Salida:** La suma de Minkowski de  $P \oplus R$

1. Triangular el polígono no convexo  $P$ , obteniendo  $t_{n-2}$  triángulos.
2. Triangular el polígono no convexo  $R$ , obteniendo  $u_{m-2}$  triángulos.
3. **Para  $i$  desde 1 hasta  $n-2$**
4.     **Para  $j$  desde 1 hasta  $m-2$**
5.          $S_k \leftarrow$  calcular  $t_i \oplus u_j$
6.     **Fin para**
7. **Fin para**

8.  $P \oplus R \leftarrow$  calcular  $\bigcup_{k=1}^{(n-2)(m-2)} S_k$

Análisis de la complejidad del algoritmo: cuando se triangulan ambos polígonos, se obtiene un conjunto de  $n-2$  triángulos  $t_i$  y otro conjunto de  $m-2$  triángulos  $u_j$ , estos pasos tienen una complejidad de  $O(n)$  y  $O(m)$  respectivamente. La suma de Minkowski de  $P$  y  $R$  es ahora la de los pares  $t_i, u_j$ . Cada suma  $t_i \oplus u_j$  tiene una complejidad constante, pero hay  $n m$  parejas de triángulos. Por lo tanto,  $P \oplus R$  es la unión de  $(n-2)(m-2)$  polígonos de complejidad constante. Esto implica que la complejidad total de  $P \oplus R$  será  $O(n^2 m^2)$  en el peor de los casos.

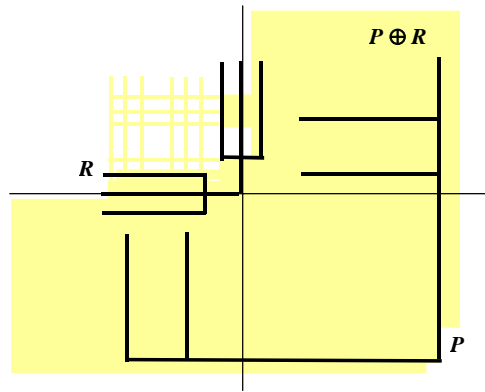


Figura 3.3: Suma de Minkowski de dos polígonos no convexos

En la figura 3.3 podemos observar dos polígonos  $P$  y  $R$  que tienen dientes horizontales y verticales, con  $n$  y  $m$  vértices respectivamente. La complejidad de  $P \oplus R$ , en este caso en particular, es  $O(n^2m^2)$ .

El siguiente teorema resume los resultados referidos a la complejidad del cálculo de la suma de Minkowski en diferentes tipos de polígonos.

**Teorema:** Sean  $P$  y  $R$  polígonos con  $n$  y  $m$  vértices respectivamente. La complejidad de la suma de Minkowski  $P \oplus R$  en el peor de los casos es:

- i.  $O(n + m)$  si ambos polígonos son convexos.
- ii.  $O(nm)$  si uno de los polígonos es convexo y el otro no convexo.
- iii.  $O(n^2m^2)$  si ambos polígonos no son convexos.

#### 4. Herramienta para el cálculo y visualización de Sumas de Minkowski.

Para cumplir los objetivos de nuestro trabajo desarrollamos una herramienta que calcula la suma de Minkowski entre polígonos, tanto entre polígonos convexos y no convexos. Esta herramienta trabaja en forma independiente de la plataforma que el usuario este usando y puede ser utilizada como herramienta de trabajo, de simulación y con fines educativos en la Geometría Computacional.

Para la herramienta se implementaron los dos algoritmos que calculan la suma de Minkowski de polígonos convexos, el algoritmo de fuerza bruta, que tiene una complejidad de ejecución de orden cuadrático, y el algoritmo mejorado con una complejidad de ejecución de orden lineal.

Para llevar a cabo la implementación de los algoritmos que calculan la suma de Minkowski fue necesario usar algoritmos auxiliares de la Geometría Computacional, que nos permitieran calcular el cierre convexo de un conjunto de puntos, encontrar los puntos extremos de los polígonos y triangular los polígonos no convexos. A continuación nombramos los algoritmos implementados en cada caso.

En el algoritmo de fuerza bruta para el cálculo de la suma de Minkowski entre polígonos convexos usamos el algoritmo conocido como Scan de Graham, que permite calcular el cierre convexo de una nube de puntos con  $O(n \log n)$  [Gra72].

En la implementación del algoritmo mejorado, usamos una variación especial del método de rotación de calibres para calcular los puntos extremos de los polígonos [Tou83].

Otro algoritmo auxiliar que implementamos es el algoritmo de Kong, que es una variación del algoritmo de “corte de orejas”, y permite triangular un polígono con un orden de ejecución  $O(kn)$  donde  $k$  es el número de diagonales y  $n$  el número de vértices del polígono [KET90].

## 5. Conclusiones y visión de futuro

Este trabajo forma parte de un proyecto de investigación iniciado por el grupo de interés en Geometría Computacional de la Universidad Nacional de San Luis conjuntamente con docentes de la Universidad Politécnica de Madrid. El principal objetivo de este proyecto es consolidar esta línea de trabajo en la UNSL, aportando nuevos enfoques y técnicas algorítmicas a las líneas de investigación ya establecidas en el LIDIC.

Uno de nuestros objetivos al comenzar nuestro trabajo, fue introducirnos en el estudio e investigación de temas relacionados con la Geometría Computacional. Podemos decir que la realización del trabajo permitió introducirnos en la disciplina y aumentar nuestros conocimientos de Geometría Computacional en general, a través del trabajo de recopilación e investigación realizado para el tema particular de este trabajo, así como también con la realización de cursos de perfeccionamiento de temas vinculados con la Geometría Computacional en sí.

Entre los objetivos planteados que se alcanzaron está el desarrollo de una herramienta que implementa la suma de Minkowski entre distintos tipos de polígonos que puede utilizarse como herramienta de trabajo, de simulación y con fines educativos en la Geometría Computacional.

Cabe destacar que, nuestra herramienta formará parte de una colección de aplicaciones que implementan algoritmos sobre distintos problemas del área de la Geometría y que desde 1988 se vienen realizando en el Departamento de Matemática Aplicada de la Universidad Politécnica de Madrid.

Como alcance y visión de futuro, se espera que este trabajo sea una etapa previa al desarrollo de una herramientas que puedan realizar otro tipo de operaciones algebraicas entre diferentes tipos de polígonos. También se plantea el desarrollo de otras herramientas que implementen sumas de Minkowski entre conjuntos de polígonos y sumas de Minkowski en tres dimensiones, ya que, por ejemplo, en el problema de planificación de movimientos el robot puede ser un poliedro.

Se espera que los autores puedan continuar la investigación de éste y otros temas relacionados a la Geometría Computacional como futuros estudios de postgrado.

**Agradecimientos:** Queremos agradecer a los árbitros anónimos que con sus comentarios ayudaron a mejorar esta presentación.

## Referencias Bibliográficas

- [Abe00] Abellanas Oar, M. *Descubriendo la Geometría Algorítmica*, 2000.  
<http://www.dma.fi.upm.es/mabellanas/divulgación/GeometriaAlgoritmica.html>

- [AHU74] Aho, A.V.; Hopcroft, J. E.; Ullman, J. *The design and analysis of computer algorithms*, Addison-Wesley Series in Computer Science and Information Processing, 1974.
- [AFH01] Agarwal, P.K.; Flato, E.; Halperin, D., *Polygon Decomposition for efficient construction of Minkowski sums*, Computational Geometry: Theory and applications N° 21, (39-61), 2001.
- [BKOS97] de Berg, M; Kreveld, Overmars, M; Schwarzkopf. *Computational Geometry: algorithms and applications*, Springer Verlag, BH 1997
- [BS01] de Berg, M; Van der Stappen, *On the fatness of Minkowski sums*, Information Processing letters, N° 81, (259-264), 2001
- [Gra72] Graham, R.L. *An efficient algorithm for determining the convex hull of a finite planar set*, Information Process Letters, 1:132-133, 1972
- [GTBH01] Gagliardi, E.; Taranilla, M.T; Berón, M.; Hernandez Peñalver, G., *La Geometría Computacional a nuestro alrededor*. Workshop de Investigadores en Ciencias de la Computación WICC 2002. Bahía Blanca, 2002.
- [HCC98] Heywood I, Cornelius S., Carver S., *Geographical Information Systems*, Addison-Wesley Longman, New York, 1998.
- [KET90] Kong, Xianshu; Everett, H.; Toussaint, G. T., *The Graham scan triangulates simple polygons*, Pattern Recognition Letters, vol. 11, November 1990, pp. 713-716.
- [KT02] Kavka, G.; Taranilla, M.T, *Implementación de una herramienta para el cálculo y visualización de sumas de Minkowski*. UNSL, 2002.
- [Lat91] J.C. Latombe, *Robot Motion Planning*, Kluwer Academic Publisher, Boston, MA, 1991.
- [Li94] Li, Zhenyu *Compaction Algorithms for Non-Convex Polygons and their Applications*, tesis doctoral, Universidad de Harvard, 1994
- [PS85] Preparata,F.; Shamos,M. *Computational Geometry: an Introduction*, Springer Verlag, NY 1985.
- [Ser82] Serra J., *Image Analysis and Mathematical Morphology*, Academic Press, New York, 1982.
- [Ser88] Serra J., *Image Analysis and Mathematical Morphology, Vol II: Theoretical Advances*, Academic Press, New York, 1988.
- [Tou83] Toussaint, G.T. *Solving geometric problems with Rotating Calipers*, Proceedings of IEEE MELECON'83, Athens, Greece, May 1983, pp. A10.02/1-4.
- [Tou85] Toussaint, G.T. *Computational Geometry*, Edited by North-Holland, Amsterdam, 1985 .
- [Tou92] Toussaint, G.T. *What is computational geometry?* Proc. IEEE, vol. 80, No. 9., (1347-1363),1992.