

“Procedimientos para obtener objetos geométricos bidimensionales”

MSc. Iván Pérez Mallea, Dr. Arsenio Iznaga Benítez

Departamento Gráfica de Ingeniería.

Instituto Superior Politécnico *José Antonio Echeverría*.

Calle 114 esq. 127, Marianao 15, Ciudad de la Habana, Cuba

Telefono: (537) 260 2267 Fax: (537) 267 1644

E – mail: iznaga@mecanica.ispjae.edu.cu.

mallea@mecanica.ispjae.edu.cu

Resumen

En el presente trabajo se demuestra que el conocimiento de la teoría de conjunto y el álgebra booleana no bastan para obtener objetos geométricos bidimensionales. Se estudian los diferentes aspectos que componen la operación booleana, presentando procedimientos y algoritmos que permiten obtener los resultados geoméricamente correctos. Se describen los pasos fundamentales de la Selección, Procesamiento y Resultados en la unión de polígonos simples.

En el proceso de selección de objetos se analizan los problemas más frecuentes en las entidades trazadas utilizando la computadora. Se estudian las características fundamentales que deben poseer dos objetos geométricos para aplicar cualquier operación booleana. Se presenta como resultado un algoritmo que permite determinar cuando los objetos seleccionados están bien definidos.

Se describe el procesamiento de los vértices que constituye el contorno de los objetos a través de lenguaje natural explicando en cada caso los aspectos más importantes. Por último, con el análisis y la discusión de los resultados del algoritmo propuesto se llega a la conclusión que se necesitan nuevos procedimientos y algoritmos para obtener los resultados geoméricamente correctos.

Introducción

El desarrollo de la expresión gráfica en computadoras ha estado marcado por el desarrollo del Hardware de trabajo y la implementación de métodos matemáticos para las soluciones geométricas. Producto de este desarrollo surge la Modelación geométrica la cual abarca un área frecuentemente llamada *geometría computacional* y se extiende a nuevos campos del modelado creando una elegante síntesis de la geometría y la computación.

El término modelado geométrico se refiere a la colección de métodos usados para definir la forma y otras características geométricas del modelo que reproduce un objeto. El modelo geométrico es la representación matemática completa, inambigua e informativa de la forma física de un objeto llevado a la forma en que se puede procesar por una computadora.

Una forma de obtener modelos geométricos complejos es a través de la combinación Booleana de dos o más objetos simples. Esta combinación booleana se fundamenta en la teoría de conjuntos específicamente cuando se desean obtener nuevos conjuntos a partir de otros existentes. El procedimiento para hacer estas operaciones debe ser no visible al usuario del programa que lo aplique, por tanto se hace necesario desarrollar algoritmos que faciliten estas operaciones.

La aplicación de la teoría de conjunto no puede hacerse de forma directa, ya que los conjuntos a representar son lugares geométricos, o sea, entidades definidas por dos subconjuntos, uno que representa los puntos de contorno y otro los puntos internos. Para un mejor desarrollo del trabajo se explicará la aplicación de esta teoría en un sistema bidimensional.

Desarrollo

El estudio y comprensión del álgebra booleana permiten la creación de entidades complejas a partir de la unión, sustracción o intersección de entidades más simples. Se entiende por entidades más simples aquellas que están compuestas por polígonos.

Un polígono es una polilínea cerrada. Las polilíneas cerradas son un conjunto de puntos que definen segmentos de rectas donde el punto inicial (primer vértice) y el punto final (último vértice) coinciden. No todas las polilíneas cerradas forman un polígono, ya que el polígono es una entidad formada por rectas continuas de forma cerrada en la cual no existen puntos repetidos, ni segmentos de rectas iguales y no existe intersección entre las líneas. Ver figura 1.

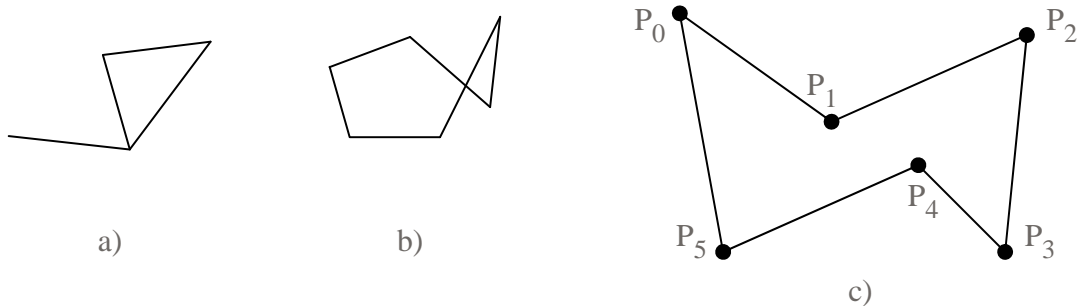


Figura 1: a) polilínea que repite la última recta, representando una sola. b) polilínea donde dos de sus rectas se cortan. c) Polilínea cerrada que constituye un polígono.

En la mayoría de los programas que se dedican al diseño gráfico existe un comando para dibujar polilíneas y otro para dibujar polígonos, solo que los polígonos que pueden ser dibujados son polígonos regulares, los polígonos irregulares continúan siendo las polilíneas. Luego la primera condición que debe cumplir un lugar geométrico plano que se desea transformar a través del álgebra booleana es que sea un polígono.

De lo anterior se concluye que para aplicar una operación booleana es necesario saber si las entidades seleccionadas son polígonos.

Proceso de selección

Antes de realizar cualquier operación booleana con polilíneas hay que realizar un proceso de selección. Esto hace que se torne general y que la condición de que una entidad seleccionada sea un polígono debe ser controlada a través de una función que devuelva verdadero o falso. Este proceso abriría una nueva lista e incluiría en ella, los lugares geométricos seleccionados por el usuario para ser procesados.

Un requisito que deben conocer los programadores es que las técnicas informáticas actuales exigen que el proceso de selección se haga con el “ratón” y se obtenga una respuesta inmediata en caso de producirse algún error.

El proceso de comprobación de la entidad seleccionada debe responder a la pregunta ¿Es una polilínea?, por ello se propone el pseudo código de la Función (ESPOL):

Función EsPol

Leer la lista de puntos

Decir que la función es FALSA

Si la lista es cerrado entonces

Analiza los vértices repetidos, si existe alguno, sale del proceso

Analiza la intersección de los segmentos, si existe alguna, sale del proceso

Decir que la función es VERDADERA

Fin de la función

La función por defecto es FALSA y solo se convierte en verdadera cuando: la lista de puntos, entiéndase la lista de la polilínea seleccionada es cerrada; no posee vértices repetidos y no existe intersección entre los segmentos de recta que la constituyen. Como se ha explicado esta función estará presente en el proceso de selección de cualquier operación booleana que se desea hacer. Por tanto, una vez conocido que las entidades a trabajar son polígonos, analicemos como se debe llevar a cabo cada proceso booleano.

Procesos booleanos

Las operaciones booleanas de UNIÓN, INTERSECCIÓN y SUSTRACCIÓN no son simples operaciones. Las reglas para su aplicación y combinación es llamada Álgebra Booleana. Las raíces de estas reglas se encuentran en la teoría de conjuntos cuando se quiere obtener nuevos conjuntos a partir de la existencia de otros conocidos.

La teoría de conjuntos permite obtener nuevos conjuntos a partir de la combinación de dos conjuntos cuales quiera expresando que: a) se puede formar un nuevo conjunto C mediante la UNIÓN de todos los elementos del conjunto A y B, se describe como $C = A \cup B$; b) se puede formar un nuevo conjunto C con los elementos comunes de A y B. Este nuevo conjunto se llama INTERSECCIÓN de A y B denotándose como $C = A \cap B$; c) se puede formar un nuevo conjunto C por los elementos de A que no son elementos de B. El nuevo conjunto se forma por la SUSTRACCIÓN de los elementos de B que existen en A, denotándose $C = A - B$.

La aplicación de esta teoría es diferente cuando los conjuntos a tratar son lugares geométricos. Para expresar matemáticamente un lugar geométrico se debe tener en cuenta que ellos están definidos por polígonos que poseen un subconjunto de puntos de contorno y un subconjunto de puntos interiores. Todo conjunto X está formado por la unión del conjunto de todos los puntos del contorno y los puntos internos esto es:

$$X = bX \cup iX$$

Donde bX; conjunto de puntos del contorno y iX conjunto de puntos interiores

El conocimiento de este fenómeno indica que en todo momento se debe dominar el área del lugar geométrico para poder realizar las operaciones. Por ejemplo: el usuario de un programa que ejecute una operación de este tipo puede que seleccione dos polígonos que se encuentran distantes uno del otro, o sea, que no se toquen ni se contengan, esto produciría un error de selección para las operaciones y no por las entidades seleccionadas. Por tanto, si no existe superposición, total o de una parte, de los lugares geométricos no puede ocurrir ninguna operación.

Por lo expresado con anterioridad, se propone una función que devuelve si dos polígonos seleccionados no se tocan, si uno contiene al otro o si al menos existe una superposición entre los lugares geométricos que definen.

El pseudo código de la función sería:

Función IntPol1Pol2

Leer los polígonos

Orientar los polígonos en sentido antihorario

Pol1 no se une con Pol2

Si todos los puntos de Pol2 están a la izquierda de Pol1, entonces

Pol1 Contiene a Pol2, sale del proceso
Si todos los puntos de Pol1 están a la izquierda de Pol2, entonces
Pol2 Contiene a Pol1, sale del proceso
Si al menos un punto de Pol2 está dentro de Pol1, entonces hay que operar
Fin de la función

Observe que esta función solo realiza la comparación de dos polígonos y en la lista de selección pueden existir más de dos, esto hace más fácil la programación de los distintos procesos como se explicará posteriormente. Una vez que la función orienta a los polígonos, toma por defecto la condición de que no se unen, cambiándola según se cumplan las condiciones que le preceden.

La operación: Unión

El proceso de unión puede realizarse entre muchos lugares geométricos según haya sido la decisión del usuario, no obstante el proceso algorítmico debe hacerse recorriendo la lista de selección, en forma de pares, hasta que se termine. Por ejemplo, se han seleccionado cuatro polígonos que se encuentran en una lista como sigue: {Pol0, Pol1, Pol2, Pol3}. El proceso de unión comienza por la unión de Pol0 con Pol1, si esta no da resultado, hace Pol0 con Pol2, si no resulta continua hasta llegar al final de la lista. En caso que no se una con ningún otro polígono se extrae de la lista de selección. Si por el contrario ocurre la unión, se extraen los dos polígonos y su resultado se une con el próximo polígono de la lista de selección. El análisis de las lista culmina cuando en la lista de selección queda un polígono o cuando en la lista final del proceso existe un polígono, cualquiera de estas dos variantes es válida.

En el análisis de dos polígonos sólo se hace necesario cuando la condición de la función IntPol1Pol2 cumpla con que al menos un punto de Pol2 está dentro de Pol1. En este caso, para llevar a cabo la unión de los polígonos se hace necesario hacer un proceso de trabajo algorítmico para “preparar” los polígonos para la unión. Para explicar este proceso nos auxiliaremos de un ejemplo de unión entre dos polígonos simples que se muestra en la figura 2.

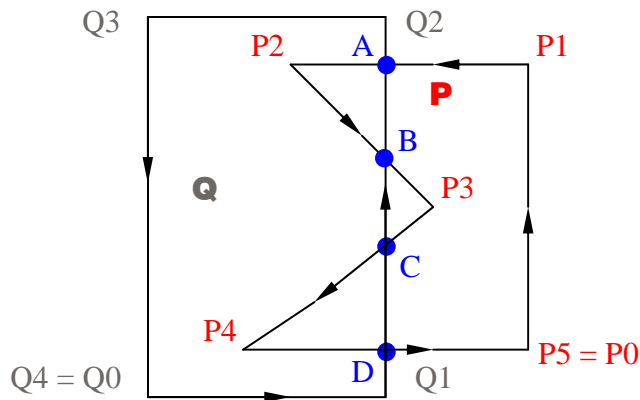


Figura 2: Representación geométrica de los

polígonos que se contienen parcialmente.

Procesamiento de los polígonos

El procesamiento se hace con el objetivo de preparar ambos polígonos para que el proceso de unión se realice satisfactoriamente. Los pasos a desarrollar en esta etapa son:

- Leer los polígonos seleccionados y orientarlos en sentido antihorario.

Q (Q0, Q1, Q2, Q3, Q4)

P (P0, P1, P2, P3, P4, P5)

- Determinar los puntos de intersección de **Q** con **P**. La determinación del punto de intersección se debe realizar tomando el primer segmento de **Q** y leyendo todos los segmentos de **P**. Donde exista la intersección, se calcula el punto agregándose este en el intervalo de recta conocido en cada polígono, quedando como:

Q1 (Q0, Q1, **A, B, C, D**, Q2, Q3, Q4)

P1 (P0, P1, **A, P2, B, P3, C, P4, D, P5**)

- En la medida en que se obtienen los puntos de intersección se hace una lista con ellos, la cual servirá para un trabajo futuro. En el ejemplo la lista será **LPI** [**A, B, C, D**]

- Para cada lista confeccionada **Q1**, **P1** se lleva a cabo el ordenamiento de los puntos de intersección en el caso en que entre dos puntos del polígono original exista más de un punto de intersección. Este ordenamiento se efectúa a través según la menor distancia que exista entre el punto del polígono y cada uno de los puntos de intersección. Esto hace que los nuevos polígonos obtenidos queden como:

Q2 (Q0, Q1, **D, C, B, A**, Q2, Q3, Q4)

P2 (P0, P1, **A, P2, B, P3, C, P4, D, P5**)

- En los nuevos polígonos obtenidos existen vértices que no son deseados. Estos vértices son los que se encuentran dentro del área de cada polígono que se une. Este paso provoca que los polígonos queden:

Q3 (Q0, Q1, **D, C, B, A**, Q2, Q3, Q4)

P3 (P0, P1, **A, B, P3, C, D, P5**)

- Una vez obtenida las listas de **Q3** y **P3** se busca si existe una secuencia mayor de dos puntos de intersección consecutivos. En el polígono **Q3** entre los puntos Q1 y Q2 tenemos la siguiente lista: **LPIS** [**D, C, B, A**]

- De esta lista de puntos de intersección consecutiva, se eliminan las que constituyen segmentos de rectas que sólo pertenecen a uno de los polígonos. Cada formación de esta lista es temporal, ya que pueden existir polígonos que posean más de una agrupación de este tipo. Una vez que sean extraídos de la lista los segmentos de recta y se formen con ellos una lista, se introducen nuevamente los puntos que quedan en **LPIS**. El proceso de eliminación de los puntos y obtención de los segmentos de recta se hace evaluando el punto medio de dos puntos de intersección consecutivos. Si el punto medio pertenece al área interior que define el otro polígono, se eliminan los dos puntos de la lista **LPIS** y se agrega a la lista de segmentos **LS**. Al aplicar este análisis queda:

LPIS [**D, A**]

LS [**C, B**]

Es importante destacar que la lista de segmentos se forma por pares de vértices. Una vez que se han extraído los segmentos de LPIS se devuelven los puntos que quedaron al intervalo de puntos en análisis. Este proceso se realiza hasta que no exista ninguna lista de puntos consecutivos mayor de 2 vértices. De aquí se obtienen nuevos polígonos y una lista de segmentos.

Q4 (Q0, Q1, **D, A**, Q2, Q3, Q4)

P4 (P0, P1, **A, B, P3, C, D, P5**)

LS [**C, B**]

Preparados los polígonos y obtenida la lista de los segmentos se puede llevar a cado el proceso de obtención de los resultados.

Obtención de los resultados

Para la formación de los resultados se toma el primer elemento de la lista **Q4** y se coloca en una lista que se denominará **LF**, en esta lista se agregan elementos de **Q4** hasta que se llega a un punto de intersección, el cual se agrega a la lista **LF** y se salta para el polígono **P4** donde se busca el valor del punto de intersección y se continua agregando valores, ahora del polígono **P4**. en la medida en que son comprobados los puntos de intersección en la lista **LPI** se van extrayendo. Esto sirve como elemento comparativo. Este primer resultado parcial termina cuando se terminen los elementos del polígono **Q4**. cuando se concluye esta etapa se obtiene:

$$\mathbf{LF} (Q0, Q1, D, P5, P0, P1, A, Q2, Q3, Q4)$$

$$\mathbf{LPI} [B, C]$$

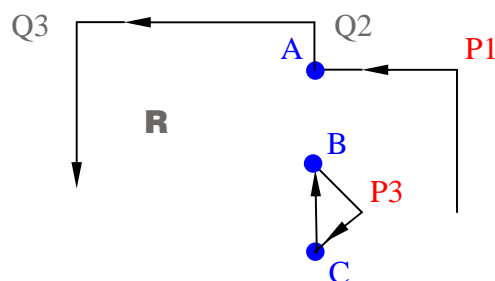
Como se observa aún las listas de los puntos de intersección no esta vacía, luego, faltan parte de los polígonos que no han sido analizadas. Por tanto, se buscar una nueva lista **LF1**, la forma más sencilla de hacerse es extrayendo de la **LPI** un punto de intersección agregándose en la lista **LF** y buscar en las listas de **Q4** y **P4** este punto de intersección. Como resultado se obtiene:

$$\mathbf{LF1} (B, P3, C, B)$$

Obsérvese que estos polígonos deben terminar con el mismo punto que comenzó. En el ejemplo ilustrado, después de tomar los puntos del polígono **P4** y llegar hasta el punto de intersección denotado con la letra **C**, se salta para el polígono **Q4**, como en este polígono no aparece dicho punto de intersección, se salta para la **LS** tomando el segmento que esta definido por esta letra. Este proceso de búsqueda de polígonos **LF** termina cuando **LPI** queda vacío, como en este ejemplo. No obstante aún no sea obtenido el resultado de la operación booleana.

Si todo el proceso anterior se ha realizado correctamente, en todas las **LF** encontradas debe existir sólo una orientada en sentido antihorario, la cual constituye el contorno del resultado de la unión. El resto de los polígonos estarán en sentido horario y constituyen los agujeros que posee el lugar geométrico. Los polígonos obtenidos en las **LF** deben ser “limpiados”, o sea, eliminar los vértices repetidos que existen dentro del intervalo sin considerar los extremos. Como se puede observar de la unión de dos polígonos simples se puede obtener una superficie compuesta por más de un polígono. En el ejemplo desarrollado se obtiene como respuesta la figura 3 y el resultado expresado en forma numérica seria:

$$\mathbf{R} \{[Q0, Q1, D, P0, P1, A, Q2, Q3, Q4] [B, P3, C, B]\}$$



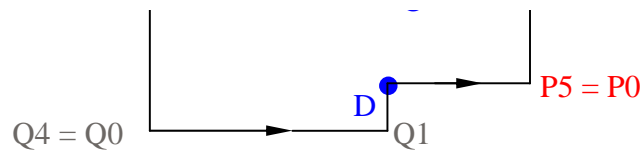


Figura 3: Lugar geométrico obtenido por la unión de dos polígonos simples

Conclusiones

Como resultado de este trabajo hemos obtenido un método robusto para la elaboración de polígonos complejos a partir de polígonos simples, que es aplicable a cualquier tipo de polígono; así como una técnica de selección aceptable para aplicaciones CAD.

Los algoritmos descritos en lenguaje natural se adaptan a una estructura de datos flexible y posible de implementar en cualquier lenguaje de programación.

Bibliografía

1. Bogoliúbov, S. Dibujo Técnico. Editorial MIR. Moscú. 1988
2. Gráfica por Computadoras. Teoría y Problemas. 1990
3. Mortenson, M. E. Geometric Modeling. Editorial John Wiley & Sons, Inc. 1985
4. Scheid, F. Introducción a la ciencia de las computadoras. Editorial Dossat, S. A. Libros McDraw-Hill. 1972
5. Foley, J. D. et al. Computer Graphics in C. Editorial Addison – Wesley Publishing Company. 1992.
6. Rodriguez, O. Dibujo Mecánico para Ingenieros. Editorial ISPJAE. 1982.
7. B. Cockburn. Discontinuous Galerkin Methods : Theory, Computation and Applications (Lecture Notes in Computational Science and Engineering). Editorial John Wiley & Sons, Inc. Marzo del 2000.