

Diseño asistido por ordenador

4^a Curso Ingeniería Informática

J.C. Torres

Dpt. Lenguajes y Sistemas Informáticos

ETS. Ingeniería Informática

Universidad de Granada

TEMA 5. Modelado de sólidos

En muchas aplicaciones los elementos que se diseñan representan objetos físicos tridimensionales, de los que es necesario calcular propiedades tales como su peso, su volumen o su momento de inercia. Estos objetos no pueden modelarse con las técnicas vistas en los temas anteriores.

Este tema aborda el estudio del modelado de sólidos, que es el conjunto de métodos y técnicas utilizadas para representar sólidos. En el se aborda la problemática especial planteada en la representación de sólidos y los tres métodos de modelado más utilizados en aplicaciones comerciales: la representación de fronteras, la geometría constructiva de sólidos y los octrees.

Indice

- 5.1. Introducción.
 - 5.1.1. Modelo topológico.
 - 5.1.2. Modelo de superficies
 - 5.1.3. Propiedades volumétricas.
 - 5.1.4. Propiedades de los métodos de representación.
 - 5.1.5. Operaciones regularizadas
 - 5.1.6. Métodos de representación
- 5.2. Modelo de fronteras (b-rep)
 - 5.2.1. Representación
 - 5.2.2. Cálculo de propiedades
 - 5.2.3. Edición
 - 5.2.4. Propiedades formales
- 5.3. Modelos de descomposición
 - 5.3.1. Enumeración
 - 5.3.2. Octrees
 - 5.3.3. Almacenamiento
 - 5.3.4. Búsqueda
 - 5.3.5. Operaciones booleanas
 - 5.3.6. Propiedades
- 5.4. Modelos constructivos: CSG
 - 5.4.1. Representación
 - 5.4.2. Visualización
 - 5.4.3. Propiedades

5.1 Introducción

Muchas aplicaciones (CAD/CAM) necesitan trabajar con elementos volumétricos con los que se puedan calcular propiedades, como por ejemplo el volumen, tamaño, tensiones, temperaturas y realizar operaciones como unir objetos, calcular su intersección, ver colisiones de elementos (robot, objeto), etc. En otros casos, el objeto que se diseña se debe fabricar mediante algún proceso automático, siendo fundamental el garantizar que representa realmente a un sólido.

El *modelado de sólidos* es el conjunto de teorías, técnicas y sistemas orientados a la representación “completa en cuanto a información” de sólidos. Dicha representación debe permitir (al menos en principio) calcular automáticamente cualquier propiedad bien conocida de cualquier sólido almacenado [Requi83]. Con los sólidos representados necesitaremos, además de visualizarlos y editarlos, calcular sus propiedades físicas (por ejemplo su peso o su centro de gravedad), y simular sobre ellos procesos físicos (como la transmisión de calor en su interior).

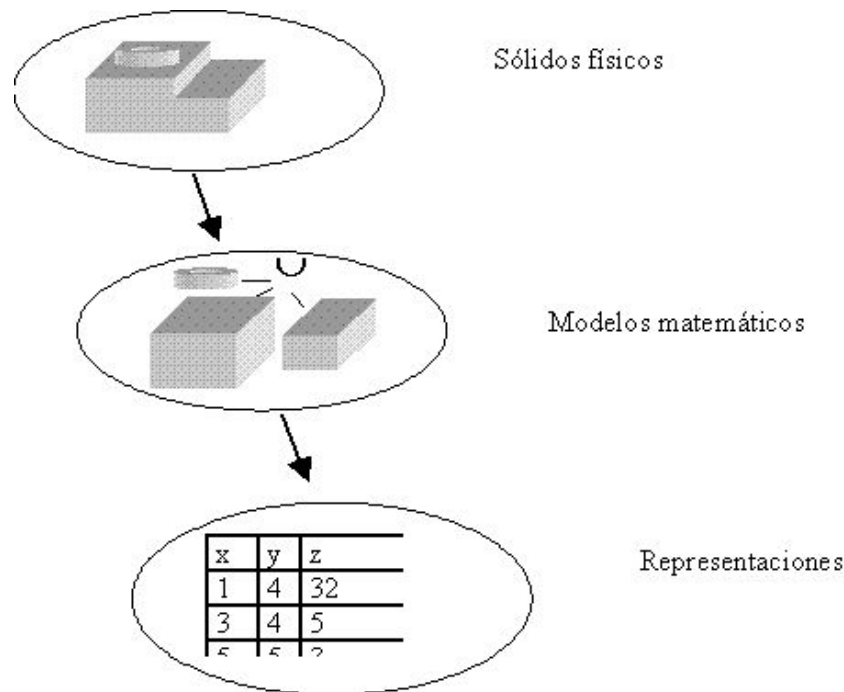


Figura 5.1. Niveles de representación

Antes de plantearnos como realizar la representación, es necesario concretar cuales son los objetos a representar. Es decir, formalizar lo que entenderemos por sólido [Mäntyla 88]. Hay dos aproximaciones diferentes al problema, una que caracteriza al sólido como un conjunto de puntos 3D, conocida como **modelo topológico** o de conjunto de puntos, y otra que caracteriza matemáticamente al sólido a partir de la **superficie** que lo delimita, esto es, su piel o frontera. De este modo estableceremos un sistema de representación con tres niveles: **sólidos físicos**, **modelos matemáticos de sólidos** y **representaciones** [Mant88, pp.32].

5.1.1 Modelo topológico

Los sólidos están definidos en el espacio Euclídeo, E^3 , ocupando una determinada porción del espacio. Podemos por tanto considerar un sólido como un conjunto de E^3 . Obviamente no todos los subconjuntos del espacio Euclídeo son sólidos (por ejemplo un punto aislado no es un sólido). Para que un conjunto de puntos represente, a nivel abstracto, un sólido deberá satisfacer, las siguientes condiciones:

- *Es cerrado y acotado.* Esto es contiene a su frontera y ocupa una porción finita del espacio.
- *Es rígido.* Los sólidos no se modifican al trasladarlos o rotarlos. O lo que es lo mismo, dos sólidos que se diferencian tan solo en una transformación rígida son el mismo sólido.
- *Homogeneidad tridimensional.* El sólido debe ser tridimensional en todos sus puntos, es decir, en cualquier punto de él debe ser posible en tres direcciones ortogonales.

Esta última propiedad se suele caracterizar mediante el concepto topológico de *regularización*. La regularización de un conjunto se define como la clausura de su interior:

$$r(S) = c(i(S))$$

El interior de un conjunto es el subconjunto de sus puntos que no están en su frontera. La clausura de un conjunto es el conjunto formado por todos sus puntos y todos sus puntos adherentes (los que están en su frontera).

Un conjunto es **regular** si es igual a su regularización

$$S = r(S)$$

Es fácil comprobar que cualquier punto o plano aislado no estará en la regularización del conjunto, es decir ser regular es equivalente a ser tridimensional en todos sus puntos. Un conjunto regular acotado se denomina un **r-set** (ver figura 4.2).

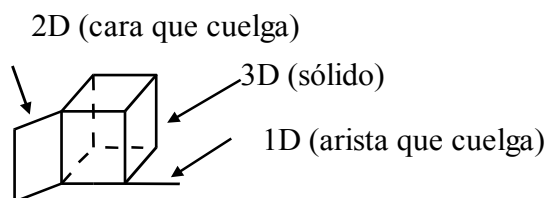


Figura 5.2. El objeto no es regular por tener elementos 2D y 1D.

5.1.2 Modelo de superficies

El modelo algebraico describe un sólido a partir de su frontera. (conjunto de superficies que separa el sólido de la parte del espacio no ocupada por él). La frontera se puede ver como la *piel* del sólido. Obviamente cualquier superficie no determina un sólido. Para que un conjunto de superficies describan un sólido deben satisfacer la siguiente propiedad

Encierra un volumen. La piel es cerrada, orientada y completa. Que la piel sea cerrada y este orientada permite determinar si un punto está dentro del sólido, y por tanto obtener el modelo topológico.

Los dos modelos son equivalentes, con determinadas restricciones.

Para que el sólido sea representable se suele imponer una condición adicional de suavidad en su frontera, más concretamente se suele exigir que la frontera sea algebraica (o al menos analítica). Esto es, debe ser representable por un polinomio de grado finito.

Este modelo nos permite utilizar representaciones de los sólidos basadas en el almacenamiento de la frontera, que es una entidad bidimensional.

Para facilitar la representación se suele exigir, además que la frontera sea *2-variedad*, esto es, cada punto de la frontera es homeomorfo a un disco en \mathbb{E}^2 . No todos los sólidos satisfacen esta propiedad, por ejemplo, dos pirámides unidas por un vértice no son 2-variedad (*2-manifold*).

5.1.3 Propiedades volumétricas

La mayor parte de las propiedades del sólido se pueden expresar como integrales de volumen de la forma:

$$\text{Propiedad}(\text{Sólido}) = \iiint_{\text{Vol}(\text{Sólido})} G_p(x, y, z) dV$$

La función a integrar, G_p , cambiará dependiendo de la propiedad a calcular. La siguiente tabla recoge algunas de las propiedades más usuales.

Propiedad	G_p
Volumen	1
Masa	ρ
Centro de gravedad	P/Masa
Momento de inercia(eje z)	$X^2 + Y^2$

5.1.4. Propiedades de los métodos de representación

Un **método de representación de sólidos** establece una relación entre sólidos y descripciones de éstos. Las representaciones de sólidos tiene los siguientes atributos y propiedades formales [Requi80]:

Dominio. El **dominio del método** de representación es el conjunto de objetos representables por él. Idealmente será el conjunto de sólidos.

Validez. Una representación es **inválida** si no corresponde a ningún sólido real. Es importante que el método de representación asegure de por sí la validez de todas las representaciones que se pueden crear o que existan procedimientos automáticos para comprobar la validez de las representaciones.

No ambigüedad. Un método es **no-ambiguo** si cada representación se corresponde con un único objeto real. Es deseable que el método no sea ambiguo.

Unicidad. Se dice que existe **unicidad** en el método de representación si cada objeto real tiene, a lo sumo, una posible representación. Si el método no asegura la unicidad no es posible comprobar la igualdad de objetos.

Otras propiedades interesantes de las representaciones son:

- Ocupación de memoria.
- Facilidad de creación y edición
- Facilidad de visualización y cálculo

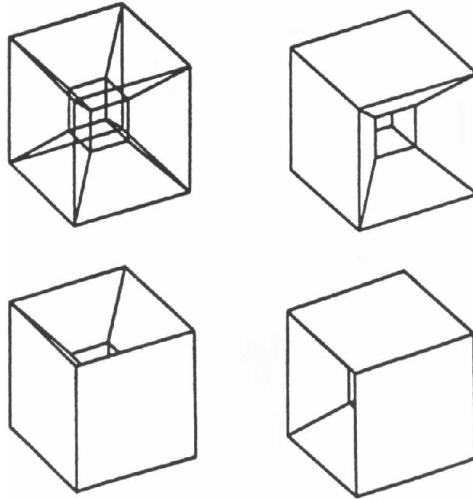


Figura 5.3. Ambigüedad del modelo de alambres

5.1.5. Operaciones regularizadas

Una de las formas más usuales, y naturales, de editar sólidos es combinarlos usando operaciones booleanas, unión, intersección y diferencia (se usa la diferencia en lugar del complemento dado que el complemento produciría un objeto no acotado). No obstante, las operaciones booleanas, tal como se definen habitualmente, no son internas al conjunto de sólidos. Es decir, pueden producir, como resultado, objetos que no sean sólidos.[Fole90; pp.535]. Para resolver este problema se utilizan operaciones booleanas regularizadas. Las operaciones regularizadas se define como la regularización del resultado de la operación, y se denotan con un asterisco.

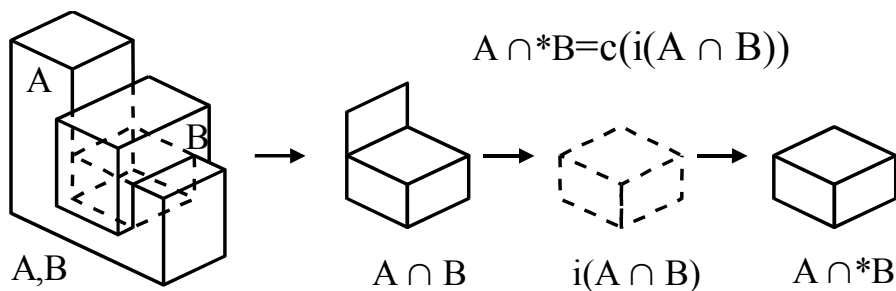


Figura 5.4. Intersección regularizada

$$A \cap^* B = r(A \cap B) = c(i(A \cap B))$$

$$A \cup^* B = r(A \cup B) = c(i(A \cup B))$$

$$A -^* B = r(A - B) = c(i(A - B))$$

El conjunto de sólidos con las operaciones regularizadas constituye un álgebra de Boole

5.1.6. Métodos de representación

Se han propuesto un gran número de esquemas de representación para sólidos. En esta sección se hace una breve introducción a algunos de estos métodos. En el resto del tema se verán con más detalle tres de estos métodos.

Instanciación de primitivas. El modelo se representa como un conjunto de primitivas que se instancian en el espacio. Para poder realizar cálculos es necesario que las diferentes instancias no se solapen en el espacio. El dominio depende del repertorio de primitivas.

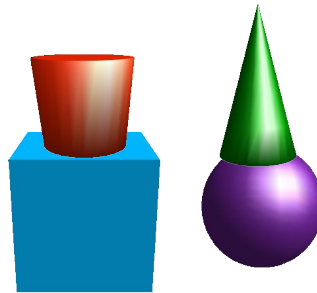


Figura 5.5. Instanciación de primitivas

Barrido (sweep). Se puede ver como una generalización de los métodos de desarrollo de superficies a partir de curvas. El elemento a desplazar es ahora una superficie, que describe un volumen al desplazarse a lo largo de una trayectoria o al rotarse.

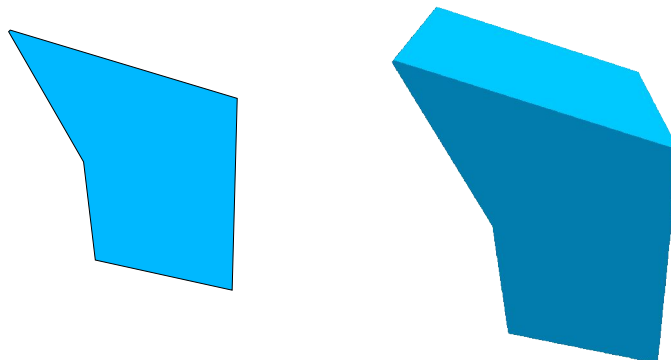


Figura 5.6. Construcción de sólido por barrido

Descomposición. El sólido se describe en base a una descomposición del espacio en una colección de elementos simples disjuntos, celdas (por ejemplo tetraedros o cubos). La representación de un sólido es el conjunto de celdas que ocupa.

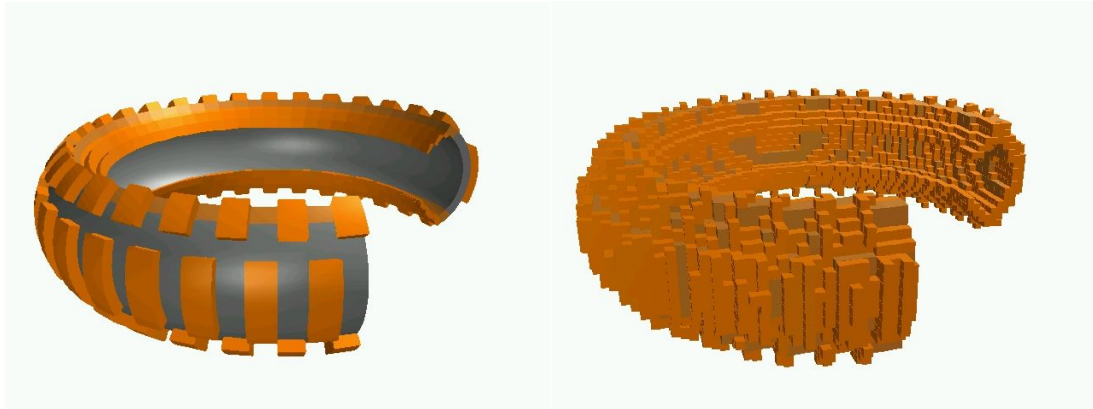


Figura5.7. Enumeración espacial

Fronteras. El sólido se representa mediante un conjunto de caras que describen su frontera. La mayor parte de los sistemas utilizan representaciones de fronteras, con caras poligonales.

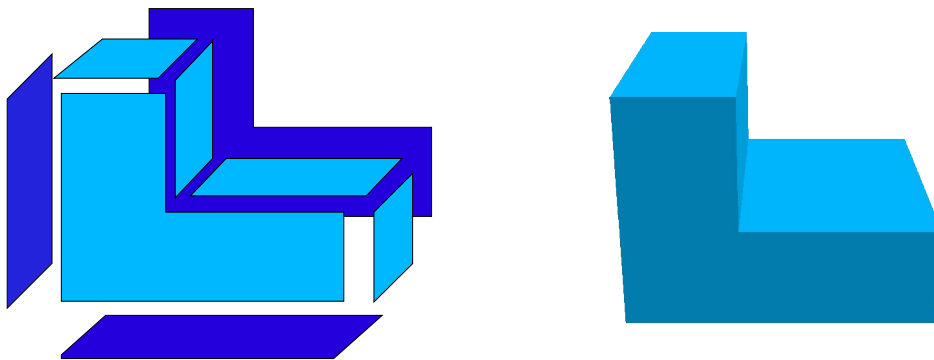


Figura 5.8. Representación de fronteras

CSG. Se almacena una expresión booleana que construye el sólido a partir de primitivas simples.

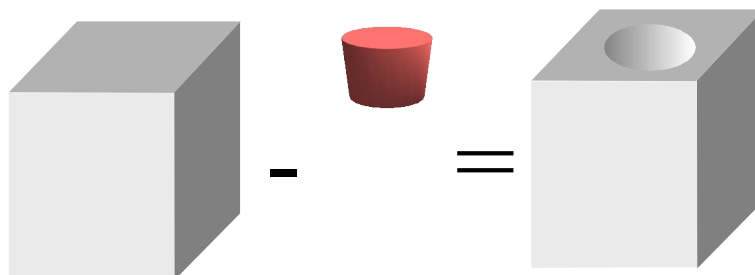


Figura 5.9. Construcción de un sólido CSG

ASM. El *modelado analítico de sólidos* describe los sólidos a partir de mallas 3D de puntos de control, combinados usando funciones mezcla, generalizando los métodos de diseño de superficies.

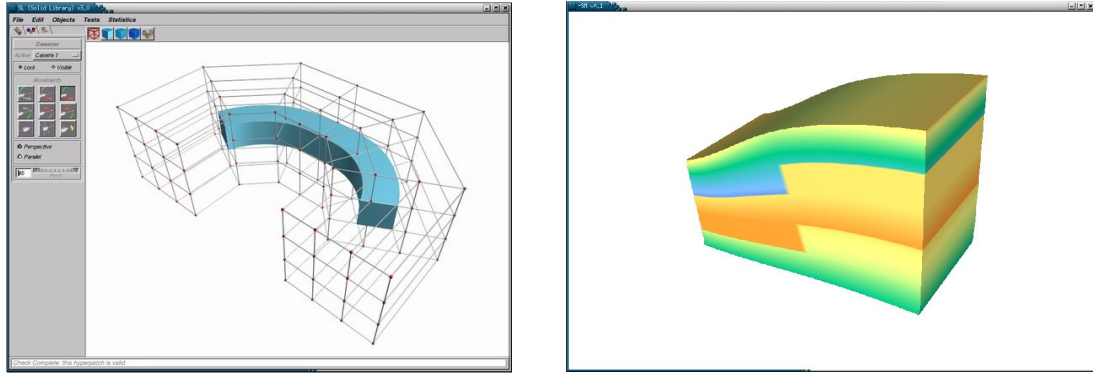


Figura 5.10. Modelado analítico de sólidos

5.2. Representación de fronteras

El método de representación de fronteras (b-rep, de “*boundary representation*”) se basa en la descripción algebraica de los sólidos, asumiendo que están delimitados por un conjunto de caras, que pertenecen a superficies orientables y cerradas. La orientación implica que es posible distinguir la cara exterior de la interior al sólido. La orientación suele estar indicada por el vector normal a la superficie.

El método surge a partir de los modelos poliédricos usados para la representación de objetos con eliminación de líneas y caras ocultas. Se puede ver como una mejora de modelo para poder describir sólidos incluyendo toda la información de la superficie. En él los sólidos se describen dando la geometría de su superficie (frontera), normalmente formada por un conjunto de caras planas, y las relaciones topológicas existentes. Para ello se distingue entre entidades geométricas (puntos, curvas, superficies) y topológicas (vértices, aristas y caras). La superficie de una cara es el plano que sustenta la cara.

Las relaciones topológicas incluyen relaciones de vecindad y de pertenencia, como por ejemplo:

- Las caras están delimitadas por aristas.
- Las aristas están delimitadas por dos vértices.
- Cada arista es compartida por dos caras.
- En cada vértice confluyen varias aristas (al menos tres)
- Vecindad entre caras.

Cuando las caras permitidas por el sistema son polígonos el conjunto de objetos representable es el conjunto de poliedros. En este caso, las caras deben ser polígonos cualesquiera, incluyendo polígonos cóncavos y con agujeros (de otro modo el conjunto de presentaciones no sería cerrado bajo operaciones booleanas). Por este mismo motivo se suele considerar como un único sólido un conjunto de poliedros separados.

5.2.1. Representación

Existen diversas representaciones que permiten almacenar modelos de fronteras. Su complejidad varía en función de las relaciones topológicas almacenadas. La opción más simple es no almacenar ningún tipo de relación, con lo que la

representación puede ser una simple lista de caras. No obstante, esta aparente simplicidad en la estructura de datos da lugar a procesos complejos para realizar cualquier operación con el modelo, desde la comprobación de validez hasta la edición o cálculo.

A modo de ejemplo, supongamos que deseamos comprobar que la frontera es cerrada (condición indispensable para que la representación sea válida). Tendremos que comprobar que en cada arista confluyen dos caras con la misma orientación. Para ello tendremos que comparar cada par de puntos consecutivos de cada cara con los pares de puntos consecutivos del resto de las caras. Este proceso es complejo (depende del cuadrado del número de caras) e impreciso, la comparación de puntos puede generar errores numéricos.

Obsérvese que esta es la estructura habitualmente usada en los sistemas que dibujan poliedros, cuando no interesa editar el objeto como un sólido, ni que represente un sólido válido, ni calcularle propiedades. Por otra parte, esta estructura es muy redundante, los puntos asociados a cada vértice se almacenan en la lista de puntos de cada cara que comparte el vértice. La existencia de información redundante conlleva la utilización de un mayor espacio, y puede ser origen de inconsistencias. La siguiente tabla muestra la representación de las dos caras visibles del poliedro de la figura 5.11.

Cara	Vértices					
1	(0,0,0)	(0,2,-1)	(0,3,2)	(0,1,2)	NULL	
2	(0,2,-1)	(2,3,-1)	(2,4,1)	(0,3,2)	NULL	
3						

Una alternativa simple es almacenar el modelo en una estructura jerárquica heterogénea, en la que la información asociada a cada cara sea una lista de aristas, y la normal a la cara. De cada arista se almacenen los dos vértices que la delimitan, y para cada vértice el punto en el que se encuentra. La estructura se puede completar con relaciones hacia arriba (de vértices a aristas, y de estas a caras). De este modo se eliminan las redundancias y se almacenan parte de las relaciones topológicas. Las superficies de las caras y las líneas de las aristas se pueden calcular cuando sea necesario a partir de los puntos que las delimitan.

Cara	Aristas					
1	0	1	2	3	NULL	
2	4	5	6	1	NULL	
3						

Aristas	V _i	V _f	C _{izq}	C _{der}
0	0	1	1	3
1	1	2	1	2
2	2	3	1	4
3	3	0	1	5
4	1	5	2	3
5	5	4	2	6
6	4	2	2	4

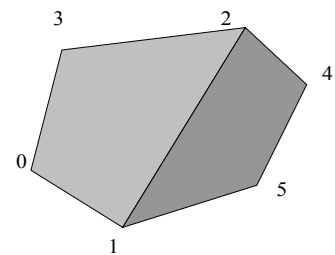


Figura 5.11. Poliedro mostrando dos caras visibles

Vértice	X	Y	Z
0	0	0	0
1	0	2	-1
2	0	3	2
3	0	1	2
4	2	4	1
5	2	3	-1

Para poder utilizar caras con agujeros es necesario complicar la estructura, almacenando para cada cara una secuencia de ciclos de aristas, de los que el primero es el perímetro exterior, y el resto agujeros practicados en él. Otra alternativa es utilizar falsas aristas, o aristas puente, para unir los perímetros interiores al exterior.

Aristas con alas

La estructura de datos más popular para almacenar un modelo de fronteras es la de aristas aladas ("winged-edge") [Manty88, Baumgart72]. Esta estructura incorpora más relaciones topológicas y organiza la jerarquía partiendo de las aristas, lo que simplifica el tratamiento de caras con agujeros¹.

El elemento central de estructura es la arista, en lugar de la cara. Cada arista almacena información de sus relaciones de vecindad con aristas y caras. Concretamente, cada arista contiene, además de su identificador:

- Sus dos vértices
- Las dos caras que comparten arista
- Las cuatro aristas que parten de ella, en las caras que la comparten.

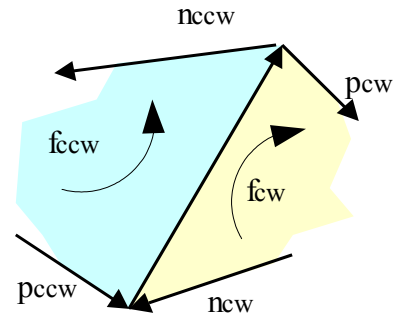


Figura 5.12. Etiqueta de aristas.

Las dos caras de la arista se etiquetan en función de como se recorra la cara cuando se va del vértice inicial al final de la arista, en **clockwise** y **counterclockwise**. La figura 4.12 muestra la información asociada a cada arista. El recorrido de los vértices se realiza en sentido anti-horario. Las tablas siguientes muestran parte de la información almacenada para el sólido de la figura 5.13.

Aristas	V_i	V_f	f_{cw}	f_{ccw}	p_{ccw}	n_{ccw}	p_{cw}	n_{cw}
0	0	1	1		3	1		
1	1	2	1	2	0	2	6	4
2	2	3	1		1	3		
3	3	0	1		2	0		
4	5	1		2			1	5
5	4	5		2			4	6
6	2	4		2			5	1

Vértice	X	Y	Z
0	0	0	0
1	0	2	-1
2	0	3	2
3	0	1	2
4	2	4	1
5	2	3	-1

Cara	Arista	Normal
1	0	(1,0,0)
2	4	(0,1,0)

¹ Cuando se utilizan aristas puente es necesario marcarlas para evitar dibujarlas. Con la estructura de aristas aladas esto no es necesario ya que se detectan por tener la misma cara a derecha e izquierda.

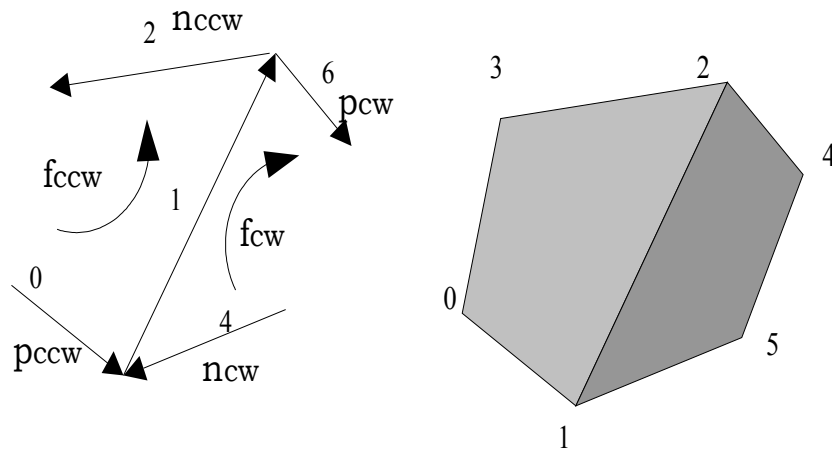


Figura 5.13. Sólido ejemplo.

Obsérvese que esta estructura no se puede utilizar para poliedros no variedad, ya que en ellos una arista podría tener más de dos caras vecinas. La alternativa para tratar este tipo de sólidos con estas representaciones es modificar el sólido para eliminar la singularidad, bien convirtiendo la arista en una unión sólida o desdoblándola (ver figura 5.14).

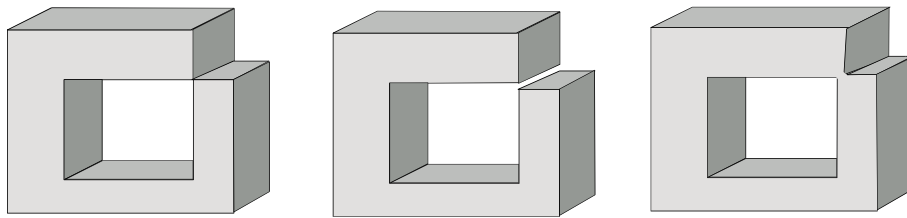


Figura 5.14. El sólido no variedad de la izquierda se puede convertir para obtener un sólido 2-variedad

5.2.2. Cálculo de propiedades

Las propiedades volumétricas se pueden calcular de forma exacta a partir del teorema de la divergencia (o de Gauss que es una generalización del de Stokes.)², que establece una relación entre la integral en volumen cerrado de la divergencia de un campo vectorial, y la integral de superficie de la componente normal del campo en la superficie del volumen. Concretamente, el enunciado del teorema es:

Teorema de la divergencia. Sea V un volumen en el espacio Euclídeo limitado por una superficie cerrada y orientable, S , y sea \mathbf{n} la normal a S , orientada hacia el exterior. Para cualquier campo vectorial \mathbf{F} , definido en V , se cumple:

$$\iiint_V (\operatorname{div} \mathbf{F}) dV = \oint_S \mathbf{F} \cdot \mathbf{n} dS$$

siendo S la superficie que encierra a V , \mathbf{n} el vector normal a dicha superficie, (obsérvese que los vectores están notados usando negrita), y div el operador divergencia.

² Apostol T.M.: "Calculus. Cálculo con funciones de varias variables y álgebra lineal, con aplicaciones a las ecuaciones diferenciales y a las probabilidades". Reverte 1980. Pp. 537-560.

Para una función vectorial \mathbf{F} , dada por

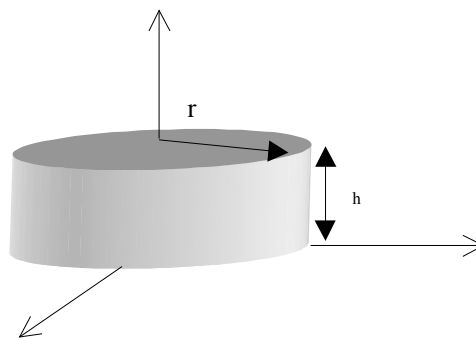
$$\mathbf{F}(x,y,z) = (F_x, F_y, F_z)$$

la divergencia se calcula como:

$$\text{div } \mathbf{F} = \left(\frac{\partial F_x}{\partial x}, \frac{\partial F_y}{\partial y}, \frac{\partial F_z}{\partial z} \right)$$

Calcular la propiedad volumétrica se reduce a una función \mathbf{F} cuya divergencia sea igual a G_p , e integrar dicha función en la superficie del sólido.

$$\text{div}(\mathbf{F}) = g$$



Ejemplo. Cálculo del peso de un cilindro. Supongamos un cilindro, cuyos eje sea paralelo al eje y, con altura h y radio de base r. Para calcular el peso se debe integrar la función constante r. Hay infinidad de funciones que cumplen esta condición, por ejemplo:

$$\mathbf{F}_a(x,y,z) = \rho (x/3, y/3, z/3)$$

y

$$\mathbf{F}_b(x,y,z) = \rho (0, y, 0)$$

En general, no es posible adaptar la función usada a la forma del sólido. En este caso es evidente, no obstante, que es más fácil integrar la segunda función.

Podemos pues calcular el peso de un objeto integrando $\mathbf{F} \cdot \mathbf{n}$ en su superficie. Si el objeto es un cilindro vertical, bastara con integrar en las dos caras horizontales, ya que el producto escalar es cero en el resto. Por tanto el volumen será:

$$\iiint_V (\text{div } \mathbf{F}) dV = \oint_S \mathbf{F} \cdot \mathbf{n} dS = \int_{\text{tapa}} y \rho dS = \pi r^2 h \rho$$

5.2.3. Edición

La edición directa, moviendo puntos de control, aunque es factible cuando se restringe el método para utilizar caras triangulares, no es aconsejable, pues es demasiado laborioso.

Los métodos de edición usuales en modeladores que trabajan con modelos de fronteras son:

- La combinación de instanciación de primitivas prediseñadas con operaciones de barrido sobre las caras.
- Las operaciones booleanas
- La modificación local del modelo por biselado de las aristas o vértices.

Hablaremos brevemente de cada una de estas opciones. Las operaciones booleanas usadas son la unión, la intersección y la diferencia. No se usa el complemento, ya que produce sólidos no válidos. Para realizar una **operación booleana** entre dos sólidos se sigue el siguiente proceso:

a) Clasificación de caras

1. Calcular todas las líneas de intersección de las caras de A con las caras del sólido B y viceversa.
2. Dividir las caras que están partidas por estas líneas. Para ello se unen las aristas consecutivas entre sí para formar nuevas caras.
3. Clasificar las caras de cada sólido respecto al otro sólido en externas e internas.

b) Selección de caras

4. En función de la operación a realizar se clasifican las caras eligiendo las que forman parte del sólido resultado.

Operación	Caras
$A \cup B$	Caras de A externas a B y caras de B externas a A
$A \cap B$	Caras de A internas a B y caras de B internas a A
$A - B$	Caras de A externas a B y caras de B internas a A

c) Crear representación

5. Reconstruir las relaciones entre elementos

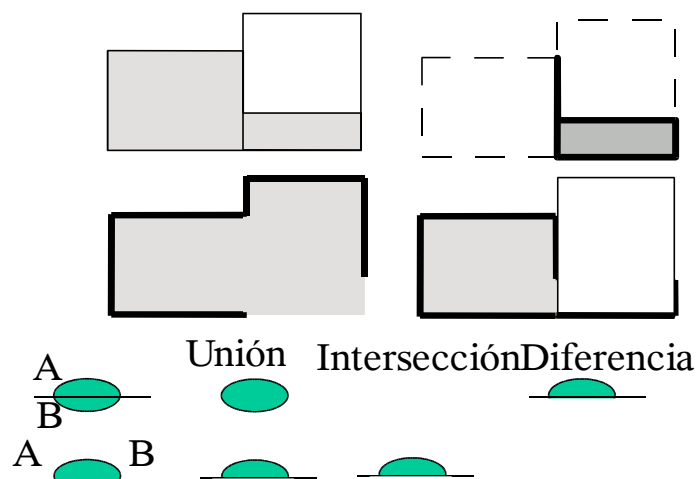


Figura 5.15. Tratamiento de caras coplanarias al realizar operaciones booleanas

Se debe tener un cuidado especial al tratar las caras coplanarias, realizando la clasificación de las caras teniendo en cuenta la orientación de las dos caras. La figura 5.15 muestra el resultado que se obtendría con dos sólidos a los que coincide una cara si esto no se tiene en consideración, y como se deben clasificar dos caras coplanarias con la misma y distinta orientación en cada uno de las operaciones.

El **biselado** consiste en sustituir una arista o un vértice por una cara. Aplicando reiteradas veces se puede obtener un sólido con perfil redondeado. El sistema modifica la representación del sólido creando las nuevas caras y modificando las relaciones de vecindad.

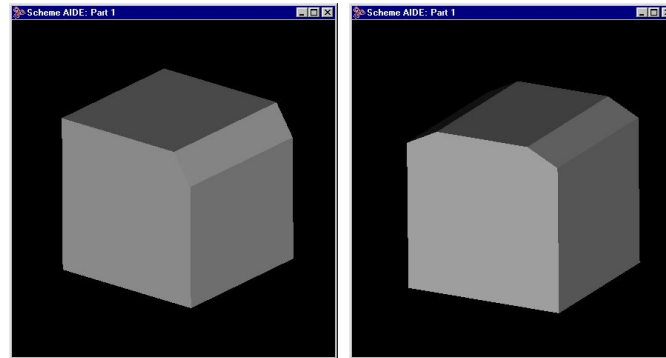


Figura 5.16. Biselado de un cubo

Una operación de **barrido** define un sólido por el volumen del espacio ocupado por una superficie cuando se desplaza a lo largo de una trayectoria, de un modo análogo a la construcción de una superficie por desplazamiento de una curva. La construcción más simple es realizar traslaciones o revoluciones, pero se pueden igualmente construir sólidos realizando un barrido generalizado, escalando y rotando la superficie mientras se desplaza a lo largo de una trayectoria arbitraria. Aún en este último caso, el conjunto de sólidos que se pueden generar es muy reducido. Por este motivo su utilización como método de representación en sí mismo está muy limitada.

No obstante es muy útil como método de modificación local de un modelo de fronteras, permitiendo su edición. De este modo se puede añadir materia a una cara del modelo realizando una extrusión a partir de la cara, o realizar una perforación haciendo una intrusión en la cara. En cualquier caso, el usuario dibuja la superficie a desplazar sobre la cara del sólido e indica el vector de desplazamiento. A partir de esta información el sistema puede generar automáticamente la información de las caras creadas (Figura 5.17).

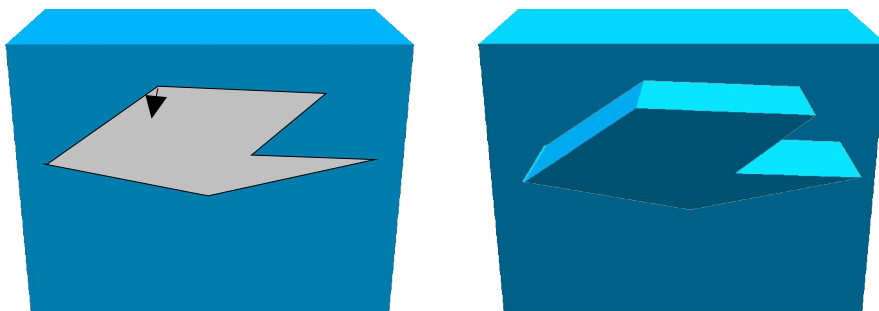


Figura 5.17. Edición por barrido

5.2.4. Propiedades de la representación

El modelo de fronteras es **no ambiguo** y su dominio es, cuando se usan caras planas, el conjunto de objetos formados por poliedros. Si se permiten caras no planas el dominio es mayor. La representación de cada sólido es **única**, salvo permutaciones de la información en las listas, si no se permiten caras coplanarias adyacentes, ya que de otro modo las caras podrían estar partidas en la estructura de datos de diversas formas.

La principal debilidad del modelo es la validez, ya que cualquier conjunto de caras, vértices y aristas no define un sólido válido. Para asegurar la validez de la representación se imponen una serie de condiciones a ésta, que pueden verse como una restricción del conjunto imagen del método. Una representación de fronteras, para representar un sólido 2-variedad válido debe cumplir:

- i) Cada arista está delimitada por dos vértices.
- ii) Cada arista separa dos caras.
- iii) Las caras que coinciden en una arista tienen orientación conforme.
- iv) Las aristas solo se intersectan en los vértices.
- v) Las caras solo se intersectan en los vértices y las aristas.

Las dos primeras condiciones se comprueban fácilmente cuando se utiliza una estructura de aristas aladas. Las tres últimas requieren la realización de comprobaciones geométricas que son costosas. Una forma rápida de detectar modelos no válidos es comprobar la fórmula de Euler, que establece una relación invariante entre caras, aristas y vértices de cualquier poliedro:

$$\text{caras} + \text{vértices} - \text{aristas} = \text{anillos} + 2 \cdot (\text{cuerpos} - \text{perforaciones})$$

$$F + V - E = R + 2(B - H)$$

Donde un anillo es una perforación entrante en una cara, una perforación es un agujero que cruza el sólido y un cuerpo es un fragmento separado del sólido. La fórmula de Euler es una condición necesaria pero no suficiente para que la representación se corresponda con un sólido válido.



$$V + F = E + R + 2(B - H)$$

Vértices	12	Aristas	18
+ Caras	8	+ Anillos	0
	20	+2·Cuerpos	2
		-2·Agujeros	0
			20

Figura 5.18. Fórmula de Euler

5.3. METODOS DE DESCOMPOSICIÓN

Representan objetos sólidos mediante la unión de un conjunto de celdas disjuntas sin agujeros.

$$S = \cup C_i$$

La representación del sólido es la lista de celdas que ocupa, que se puede codificar como un bit de ocupación por cada celda (C_i), si la geometría de las celdas es fija, o como la lista de parámetros de las celdas ocupadas. Los métodos que utilizan celdas de geometría fija suelen utilizarse para realizar representaciones aproximadas.

Estos métodos se pueden utilizar para representar sólidos heterogéneos, añadiendo simplemente el valor de propiedad de cada celda.

Los métodos de descomposición puede estructurar las celdas en una estructura jerárquica, lo que ayuda a localizar información y a compactar la representación del modelo, este es el caso de los árboles octales (o de octantes, octrees), y de los árboles de partición binaria del espacio (BSP). Cuando la representación no es jerárquica se dice que el método de representación es una enumeración de celdas.

En la siguiente sección se estudia la enumeración espacial y en la sección 3.2 los Octrees.

5.3.1. Enumeración espacial

El espacio donde reside el sólido se divide en cubos de igual tamaño, que conforman una rejilla 3D. Por cada cubo (voxel), se almacena la información de ocupación del sólido, esto es, si el voxel pertenece o no al sólido. En la mayor parte de los casos es necesario realizar una aproximación ya que el voxel no estará ni totalmente dentro ni totalmente fuera del sólido.

La estructura de almacenamiento es simplemente un array 3D de booleanos. En cierto modo, la representación es semejante a una imagen 3D del sólido. El principal inconveniente de la representación, además de ser aproximada, es que no es concisa (ocupa mucho espacio).

5.3.2. Octrees

Un octree describe, dentro de un cubo inicial, y recursivamente los octantes ocupados por el sólido. Dado un cubo que encierra al sólido, se divide en ocho octantes, que podrán estar totalmente ocupados por el sólido, vacíos o parcialmente ocupados. Para los octantes que están parcialmente ocupados se repite el mismo proceso. Así hasta alcanzar un nivel de precisión prefijado.

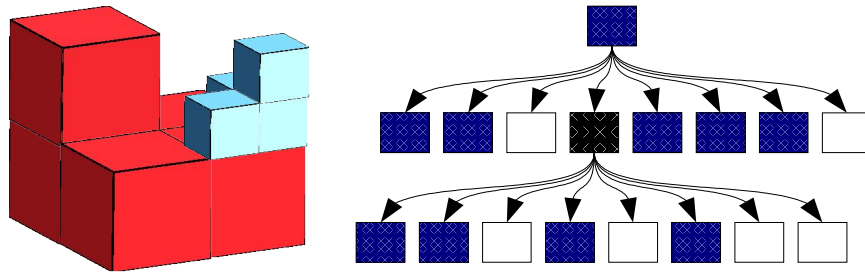
La representación puede verse como una compactación de una enumeración espacial.

5.3.3. Almacenamiento

De forma natural la representación se puede almacenar como un árbol octal, en el que cada nodo interno representa un cubo parcialmente ocupado, y los nodos terminales puede estar ocupados o vacíos. Los ocho hijos de cada nodo interno representan los ocho subcubos en que se divide el cubo.

Esta representación permite localizar de forma rápida la información de un nodo, pero requiere un espacio adicional para alojar los punteros a los hijos. Cuando no se va a operar con el

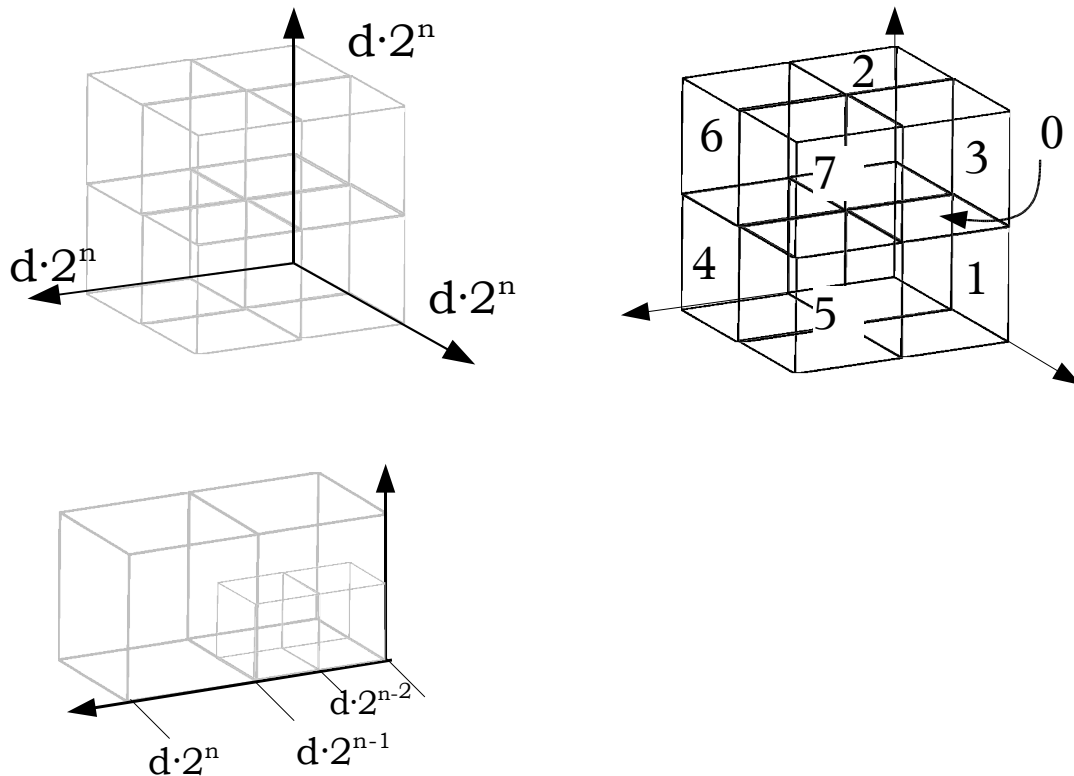
sólido, por ejemplo para almacenarlo en un fichero, suele ser preferible una codificación lineal, obtenida mediante un recorrido del árbol (por ejemplo en orden central).

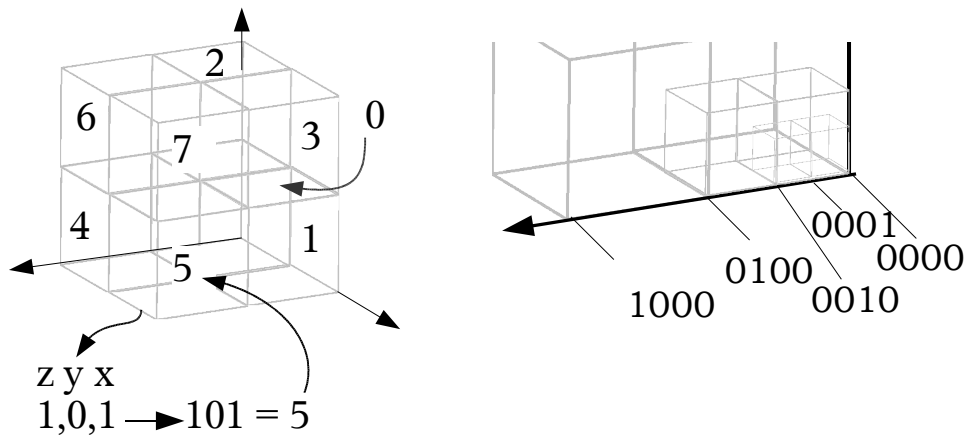


B B W (B B W B W B W W) B B B W

5.3.4. Búsqueda

La localización de un punto del espacio en el árbol, que es necesaria para la mayor parte de las operaciones, y en concreto para saber si el punto está dentro del sólido se puede realizar de forma rápida si se ha realizado el etiquetado de los nodos de forma adecuada. Concretamente se puede hacer que el índice del nodo hijo se obtenga por yuxtaposición de su posición relativa en las tres direcciones (ver figura). Asumiendo una resolución máxima conocida, esto es un tamaño conocido, d , del voxel de más pequeño a usar, podemos calcular los índices de un voxel dado en coordenadas enteras, en el rango 0 a 2^n , siendo n la máxima profundidad del árbol.





5.3.5. Operaciones booleanas

La forma más simple de editar un octree es realizando operaciones booleanas. Dados dos octrees con la misma caja englobante, las operaciones booleanas se pueden calcular nodo a nodo, realizando un descenso sincronizado en los dos operandos. Las siguientes tablas muestran el procesamiento de las operaciones booleanas, representando por \square un nodo vacío, por \blacksquare uno lleno, y por \boxtimes un nodo interno.

		B		
A	$A \cap B$	\square	\blacksquare	\boxtimes
	\square	\square	\square	\square
	\blacksquare	\square	\blacksquare	Copiar B
	\boxtimes	\square	Copiar A	Descender

		B		
A	$A \cup B$	\square	\blacksquare	\boxtimes
	\square	\square	\blacksquare	Copiar B
	\blacksquare	\blacksquare	\blacksquare	\blacksquare
	\boxtimes	Copiar A	\blacksquare	Descender

		B		
A	$A - B$	\square	\blacksquare	\boxtimes
	\square	\square	\square	\square
	\blacksquare	\blacksquare	\square	Copiar -B
	\boxtimes	Copiar A	\square	Descender

5.3.6. Propiedades

Los principales inconvenientes de la representación son su carácter aproximado y el espacio ocupado. Por otra parte, no es trivial realizar transformaciones geométricas, salvo rotaciones de múltiplos de 90 grados, escalados por múltiplos de dos o traslaciones en múltiplos del tamaño de un voxel.

Como ventajas destacan la facilidad para realizar operaciones booleanas y calcular propiedades. Las propiedades volumétricas se pueden calcular realizando una simple sumatoria sobre el conjunto de nodos terminales ocupados:

$$Propiedad(Sólido) = \iiint_{Vol(Sólido)} G_p(x, y, z) dV = \sum_{nodos\ llenos} G_p(voxel) \cdot V(voxel)$$

Se han realizado varias extensiones del método de representación para permitir una representación exacta de sólidos poliédricos. Entre estas cabe destacar los octrees extendidos, que incluyen información de la geometría de la frontera en los nodos terminales.

5.4. GEOMETRIA CONSTRUCTIVA DE SOLIDOS.

La geometría constructiva de sólidos (CSG) no representa explícitamente la geometría del sólido, si no que lo describe como una expresión booleana, en la que intervienen primitivas (sólidos básicos predefinidos).

Más formalmente, un modelo CSG se puede expresar como

$$Sólido = T_{g1}(Sólido_1) \text{ OpB } T_{g2}(Sólido_2) \mid T_{g3}(primitiva)$$

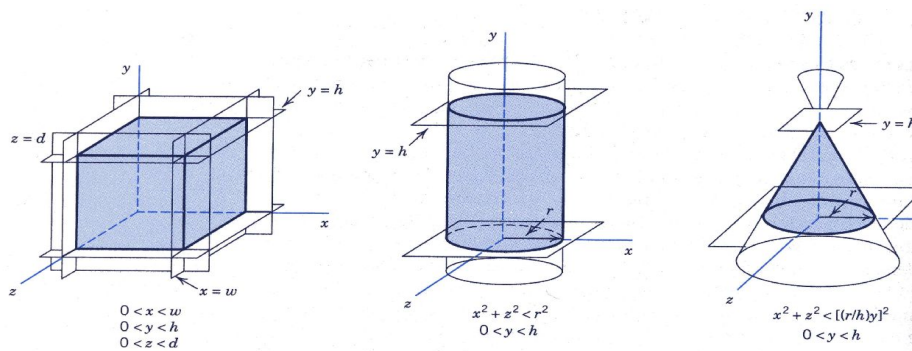
donde las T_g son transformaciones geométricas y OpB es una operación booleana (\cup , \cap o $-$).

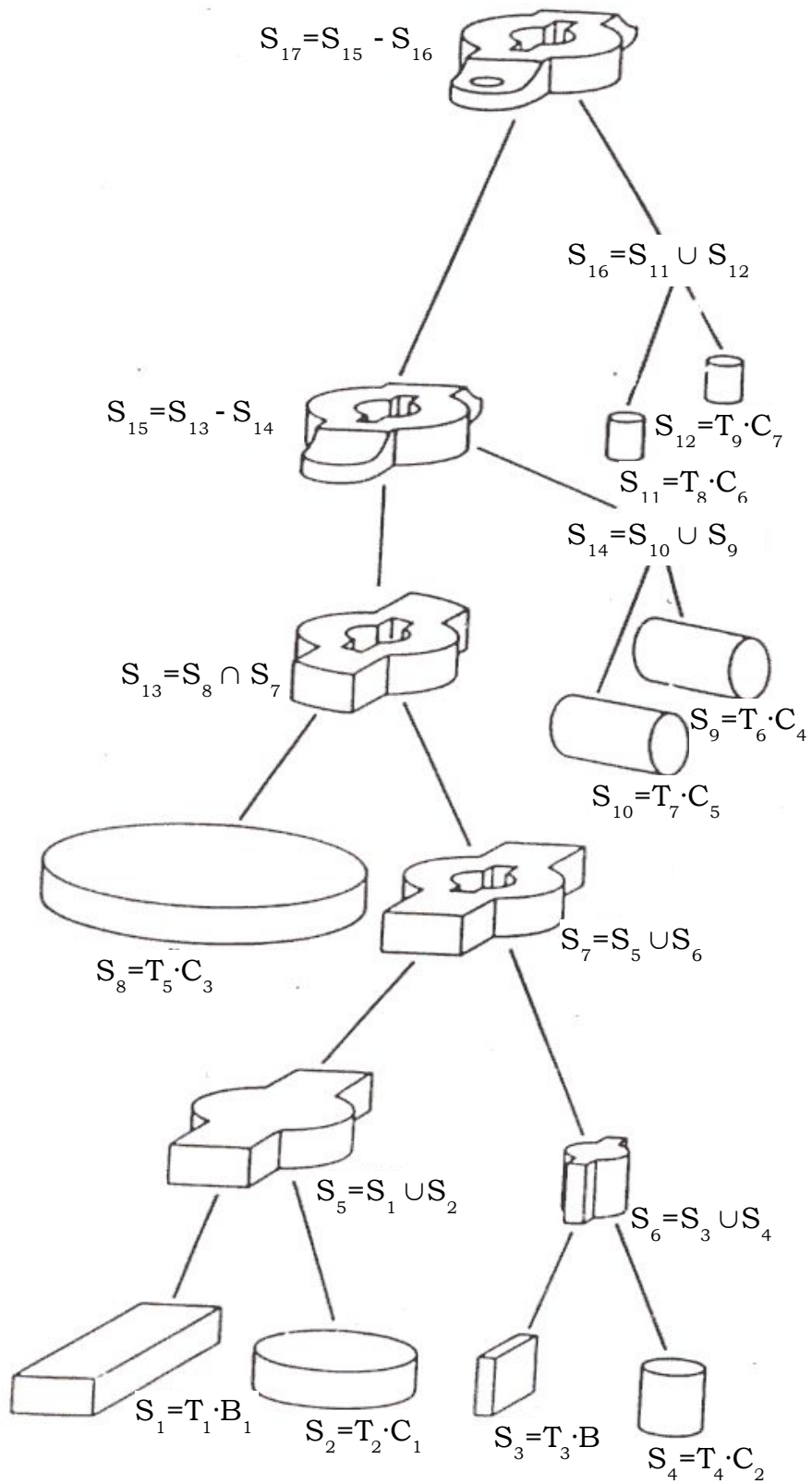
Obviamente la representación es muy concisa, pero a costa de no almacenar explícitamente la geometría, lo que hace que la visualización no sea trivial.

5.4.1. Representación

La representación usual de un modelo CGS es como un árbol binario, en el que los nodos terminales contienen primitivas y los nodos internos operadores booleanos. Asociado a cada arco hay una transformación geométrica.

Las primitivas, que puede estar parametrizadas, se representan como intersección de semiespacios, lo que permite un tratamiento homogéneo de todo el árbol.





5.4.2. Visualización

La visualización del modelo debe realizarse convirtiendolo a Brep, usando ray casting o por líneas de barrido ("scan line").

Para convertir el modelo CSG a Brep se transforman a Brep las primitivas y se evalúa el árbol CSG.

La visualización directa por trazados de rayos o líneas de barrido resuelve el problema en 1D o 2D calculando la intersección de cada primitiva con los rayos que pasan por cada pixel, o los planos que contienen cada línea de barrido.

5.4.3. Propiedades

La geometría constructiva de sólidos es un método de modelado no ambiguo, con un dominio amplio (condicionado por el repertorio de primitivas disponibles) y válido. La principal debilidad del método es la no unicidad de las representaciones, ya que hay infinidad de formas de representar cualquier sólido.³

Bibliografía:

- Fole90** Foley J.D.; van Dam A.; Feiner S.K.; Hughes J.F.: "Computer Graphics. Theory and Practice". Addison-Wesley 1990.
- Brun93** Brunet, P.; "Modelado geométrico de volúmenes" Cursos del CEIG'93
- Manty88** Mantyla, M.; "An introduction to solid modeling". Computer Science Press 1988
- Bron87** Bronsvoort W.; "Geometric Modeling" Tutorial EG'87
- Manty89** Mantyla M.; "Avanced Topics in Solid Modeling". Tutorial EG'89