

• 1400008459  
còpia 1

**Modelat sòlid:  
realització d'operacions booleanes  
amb model de fronteres**

D. Ayala

Report LSI-89-2



## Modelat sòlid: realització d'operacions booleanes amb model de fronteres.

Aquest treball presenta un algorisme per realitzar operacions booleanes amb model de fronteres. Es basa en un algorisme en 2D per polígons. L'algorisme consta de 4 passos. Els tres primers son comuns a qualsevol operació i obtenen totes les cares resultants de la interferència entre els dos objectes inicials amb una informació associada a cadascuna que indica si és interior o exterior a l'altre objecte. El quart depèn de l'operació i selecciona aquelles cares de les obtingudes que corresponguin a l'objecte final segons l'operació booleana.

This report presenta an algorithm that does boolean operations to two solids in boundary representation. It is based in a 2D algorithm that works on polygons and has 4 steps. The three firsts steps are common for all boolean operation and they obtain all the faces produced when operate the two objects. Moreover, they compute an associate information for each face that indicates if it is inside or outside the other object. The fourth step depends on the specific boolean operation and selects that faces that correpond to the final object.

UNIVERSITAT POLITECNICA DE CATALUNYA  
ESCOLA TÈCNICA SUPERIOR D'ENGINYERS INDUSTRIALS

DEPARTAMENT DE MÈTODES  
INFORMÀTICS

MODELAT SOLID: REALITZACIO  
D'OPERACIONS BOOLEANES AMB  
MODEL DE FRONTERES

Dolors Ayala Vallespí

Novembre de 1985

Informe de Recerca DMI01-85

Realitzat amb la col.laboració de  
Luis Antonio Delgado Morante

## 1.- INTRODUCCIÓ

### 1.1 Modelat Geomètric

El Disseny Assistit per Computador (C.A.D.) s'està utilitzant cada cop més en la indústria degut a que ofereix una sèrie d'avantatges com són la interactivitat i facilitat de crear nous dissenys, la simulació del comportament del sistema fins i tot abans de crear el prototip, la generació de planols de qualsevol vista amb detalls i seccions i la possibilitat de connexió amb un sistema de fabricació assistida per computador (C.A.M.) per a la mecanització automàtica d'un prototip.

El modelat geomètric és la part del CAD que cal resoldre primer. El modelat de l'objecte consisteix en definir interactivament la forma tridimensional que haurà de tenir aquest. Dins del computador hi haurà emmagatzemat un model tridimensional complet de l'objecte, que en permetrà la generació de qualsevol vista i secció i que contindrà tota la informació necessària per al càlcul de propietats geomètriques com: superfície, volum, pes, centre de gravetat, moments d'inèrcia, etc.

Per tant, una part fonamental del modelat geomètric és l'elecció del model en que vindran representats els objectes a dissenyar. En aquest sentit s'han definit unes normes que han de complir els objectes a representar i els sistemes de representació [1].

Els sistemes de representació o models més coneguts són els següents:

- model de filferros. Amb aquest model només tenim la informació de les 3 coordenades a l'espai de tots els vèrtex que té l'objecte i també coneixem totes les arestes que uneixen parells d'aquests vèrtex. Aquest és un model molt senzill però és ambigu [1].
- model de fronteres. Es una ampliació del model de filferros en el sentit de que també es guarda la informació de les cares. Aquest model ja no és ambigu i, per tant, permet dur a terme procediments com seccions, operacions booleanes, generació de vistes amb eliminació de les parts amagades i altres processos que no es podrien fer amb el model de filferros.
- model CSG. En el model CSG (Geometria Constructiva de Sòlids) el que realment guarda el computador és el procediment que s'ha seguit en la creació de l'objecte. I els objectes es creen partint d'uns

sòlids molt elementals (prismes, cilindres, cons i esferes) i obtenint el disseny desitjat a base d'aplicar-los algunes operacions com:

- \* translació, rotació i escalats (operacions de transformació)
- \* unió, intersecció i diferència (operacions booleanes)

Aquest sistema fa que la quantitat d'informació guardada per un objecte sigui molt petita, però el procés de visualització és molt costós en un model d'octrees. Es considera l'espai que conté l'objecte dins d'un cub i aquest es va subdividint recursivament en 8, analitzant, per cadascun dels octants, si és totalment interior o exterior a l'objecte (octant negre o blanc), i en aquest cas no caldrà subdividir més l'octant; o si està parcialment ocupat per l'objecte (gris) i en aquest cas cal subdividir l'octant. Aquest és un mètode que està encara en fase de recerca ja que ocupa molta memòria i fa gairebé impossible la creació d'objectes directament en aquest model. Però en canvi és el model que fa més simple el procediment de realització d'operacions booleanes [2], [3].

I a més de l'elecció del model, és important que els Sistemes de modelat geomètric puguin dur a terme processos com la creació d'objectes, visualització des de qualsevol punt de vista, càlcul de les propietats volumètriques, obtenció de seccions i realització d'operacions booleanes entre objectes ja existents.

## 1.2 El model de fronteres.

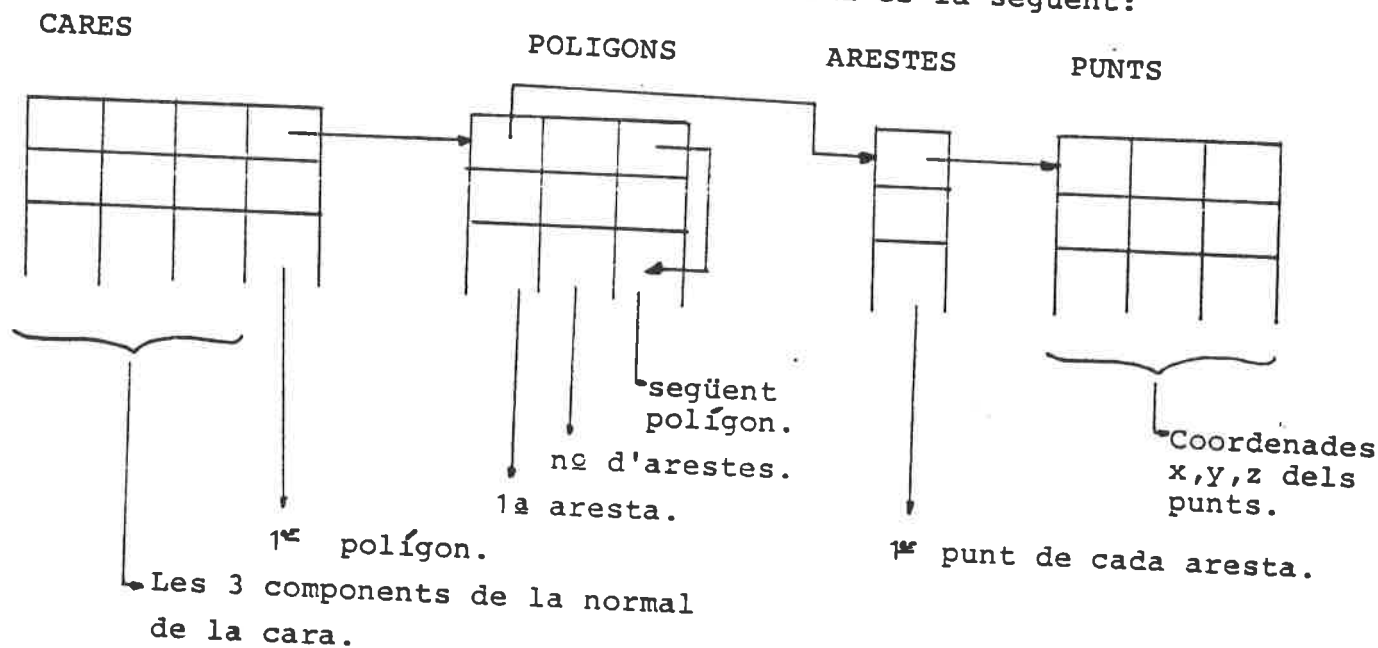
El model de fronteres, que és sobre el que es desenvoluparan les operacions booleanes que es presenten, és un dels sistemes més utilitzats en l'actualitat. Com s'ha dit abans, el model guarda informació de les cares, arestes i punts de l'objecte i així, degut a que un sòlid ve totalment definit per les seves fronteres, tenim un model de representació no ambigu que ens permet dur a terme totes aquelles operacions que s'han de realitzar en un sistema de CAD. Aquest model fa especialment simples les operacions de visualització i càlcul de propietats volumètriques, fa que la creació d'objectes sigui bastant senzilla, però en canvi fa bastant costosos els processos d'obtenció de seccions de l'objecte per un pla i el de realització d'operacions booleanes.

El model de fronteres es pot implementar de diverses maneres. A continuació s'exposa la implementació del model de fronteres sobre la

que es treballa.

El model consta de 4 estructures: cares, polígons o anells (rings), arestes i vèrtex. Abans d'entrar en més detall, potser cal definir que és el que s'entén per polígons. Una cara pot estar formada per un sol bucle o bé per diversos bucles: un d'exterior que determina la frontera de la cara i els altres interiors. Cadascun d'aquests bucles és el que s'anomena polígon i ve determinat per un conjunt d'arestes que formen un contorn tancat; i el conjunt dels polígons amb aquestes característiques: coplanars i amb un d'ells que deixi dins a tots els altres, és l'entitat que s'anomena cara.

Esquemàticament, l'estructura utilitzada és la següent:



Esquema 0. Estructura de dades del model de fronteres.

Finalment, només cal dir que aquesta estructura forma part d'un Sistema de Modelat Geomètric en el que ja es poden realitzar totes les operacions de creació, visualització, càlcul de les propietats volumètriques i obtenció de seccions i que ha estat realitzat en el Departament de Mètodes Informàtics de la E.T.S.E.I.B. [7], [8]

### 1.3 Operacions Booleanes. Sistemes existents

La realització d'operacions booleanes és un procés interessant dins d'un sistema de CAD. Entre altres, dona les següents avantatges:

- permet definir nous objectes a partir de la 'suma' o 'resta' d'altres ja existents.
- permet simular processos de fabricació com ara el fresat i el taladrat.
- permet detectar interferències o colisions espaials [4]

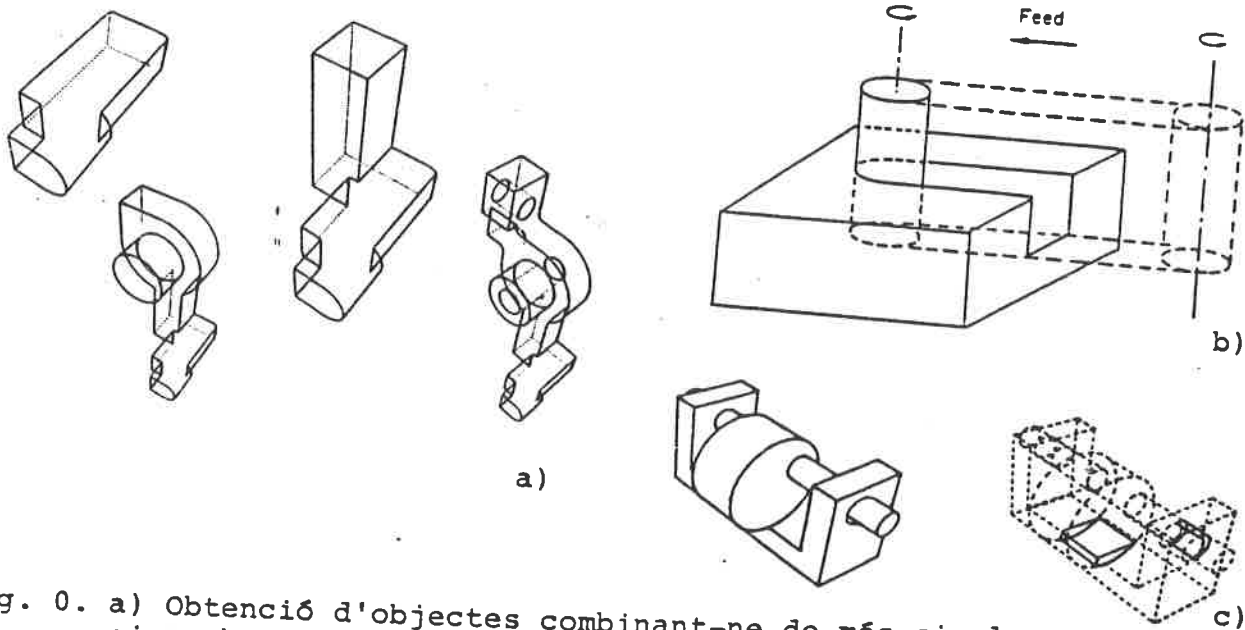


Fig. 0. a) Obtenció d'objectes combinant-ne de més simples amb operacions booleanes. b) Simulació d'una operació de fresat restant la regió d'espai escombrada per la fressadora. c) El conjunt no és realitzable ja que en un moment donat es produiria una col·lisió.

Actualment la majoria dels sistemes de modelat geomètric permeten realitzar operacions booleanes pero en canvi de pocs d'ells n'hi ha informació escrita.

Els 2 autors que han publicat sobre el tema són, per una banda, A.G. Requicha i H.B. Voelker de la Universitat de Rochester (USA) que han desenvolupat un sistema de modelat geomètric anomenat PADL, i per altra banda, M. Mantyla de la Universitat de Helsinki (Finlandia) que també ha desenvolupat un Sistema experimental de modelat geomètric anomenat GWB (Geometric Workbench).

En el primer dels 2 mètodes el problema es tracta en 2 parts. En primer lloc, cal dur a terme una classificació. S'ha de classificar un conjunt candidat  $x$  respecte a un conjunt de referència  $S$  de forma que  $x$  quedi repartit en els subconjunts  $x \cap S$ ,  $x \setminus S$  i  $x \setminus S$ . Al fer les operacions booleanes entre 2 sòlids, hi haurà dos conjunts de

referència que seran els 2 sòlids i els conjunts candidats seran les cares i les arestes de cada sòlid. Els resultats de la classificació han de tenir la mateixa dimensionalitat que el conjunt candidat: (cares si el conjunt candidat és cara i arestes si és aresta). D'aquesta forma es duen a terme les operacions regularitzades, es a dir, que el seu resultat no tindrà cap part 'dangling' (de menys de 3 dimensions) 1 .

Un cop feta la classificació, la segona part consistirà en la combinació de les classificacions. Donats 2 conjunts de referència A i B i un conjunt candidat x, coneixem ja els subconjunts  $x_{in}^A$ ,  $x_{on}^A$ ,  $x_{out}^A$ ,  $x_{in}^B$ ,  $x_{on}^B$  i  $x_{out}^B$ . A partir d'aquesta informació s'haurà d'obtenir la informació resultat  $x_{in}^S$ ,  $x_{on}^S$  i  $x_{out}^S$  essent  $S=AxB$  (x indica una operació booleana regularitzada) [4].

El segon mètode es caracteritza bàsicament per una particular estructura del model de fronteres 5 . El procés que dur a terme és bastant semblant al de la majoria que consisteix en calcular primer les interseccions entre cares dels 2 sòlids; obtenir el polígon d'intersecció format pels nous vèrtex i arestes que s'han creat a les cares dels 2 sòlids, i, finalment, classificar les cares resultants d'acord amb si seran incloses o no al sòlid resultant. Una diferència d'aquest algorisme es que en lloc de fer la comparació cara-cara inicial, compara arestes d'un sòlid amb cares de l'altre. I per formar el polígon d'intersecció utilitza els operadors de Euler [5].

El mètode que es presenta és un mètode bastant senzill obtingut a partir d'una extrapolació d'un mètode en 2D. De totes formes és bastant semblant als mètodes esmentats i el fet que sigui més senzill és que el mètode en si no tracta el cas de cares coplanàries, cas que s'elimina mitjançant un preprocés.

## 2.- DESCRIPCIÓ GENERAL DEL MÈTODE QUE ES PRESENTA

En realitzar alguna operació booleana (unió, intersecció i diferències) entre dos sòlids, ens trobem davant d'un problema d'interferència entre dos sòlids. Vist des del punt de vista del model de fronteres, en que venen representats els sòlids, la interferència dels sòlids vindrà totalment representada per la interferència de les cares d'aquests sòlids. Per tant una part fonamental del problema sera estudiar aquesta interferència.



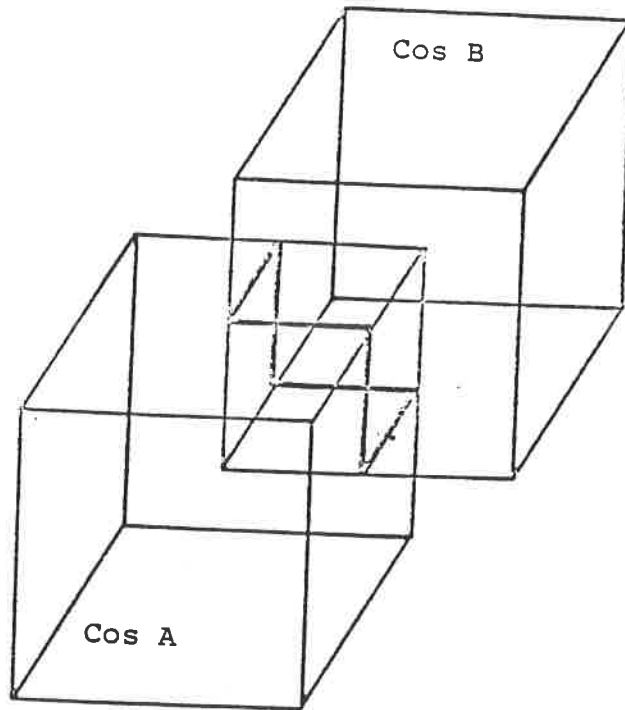


Figura 1. 2 cubs interferint

En el mètode que es presenta, el que interessa d'aquest estudi, és obtenir, per cada sòlid, totes les cares que tindrà com a resultat de la interferència. Es poden definir 2 tipus de cares: les cares que no interfereixen i que per tant romanen intactes (cares C3 del sòlid A i C4 del sòlid B) i les noves cares, que són part de les cares antigues i que han estat degudes a la interferència (les cares C1 i C2 són les 2 noves cares procedents de la cara de dalt del sòlid A, i que són resultat de la interferència d'aquesta cara amb 2 cares laterals del sòlid B. figura 1).

Per tant, el que fa en primer lloc aquest mètode, és obtenir el que en podríem dir l'explosió de cares resultat de la interferència entre els sòlids. I totes aquestes cares tenen una propietat que no tenien les cares inicials dels 2 sòlids: donada una operació booleana a fer, el sòlid final vindrà totalment representat per un subconjunt d'aquestes cares. I, per tant, el problema que ens resta ara és el de determinar quin és aquest subconjunt.

Per resoldre aquest problema, hem de conèixer dues propietats per cada cara: la primera, que és immediata, és saber de quin sòlid prové (així les cares C1, C2 i C3 provenen del sòlid A i la cara C4 prové del B); la segona propietat, que caldrà determinar, és saber, per cada cara, si és interior o no a sòlid del qual no prové. Així caldrà veure si

les cares C1, C2 i C3 són interiors o no al sòlid B i si la cara C4 es interior o no al sòlid A. Com es pot veure de la figura 1, les cares C1 i C3 són exteriors a B i en canvi la cara C2 n'és interior i la cara C4 és exterior a A.

Quan es coneixen aquestes 2 propietats per totes les cares, per triar el subconjunt d'elles que constituirà el sòlid resultant de l'operació booleana donada, només cal basar-se en la següent taula:

Op. Bool.	Cares que compleixen les següents propietats
$\cup$	$\in A \text{ i } \notin B \quad \cup \quad \in B \text{ i } \notin A$
$\cap$	$\in A \text{ i } \subset B \quad \cup \quad \in B \text{ i } \subset A$
A-B	$\in A \text{ i } \notin B \quad \cup \quad \in B \text{ i } \subset A$
B-A	$\in A \text{ i } \subset B \quad \cup \quad \in B \text{ i } \notin A$

Taula 1. Operacions Booleanes

Aquí, el símbol  $\in$  vol dir 'prové de' ( $\in A$  vol dir prové de A) i el símbol  $\subset$  vol dir 'és interior' ( $\subset A$  vol dir és interior al sòlid A i  $\notin B$  vol dir és exterior al sòlid B)

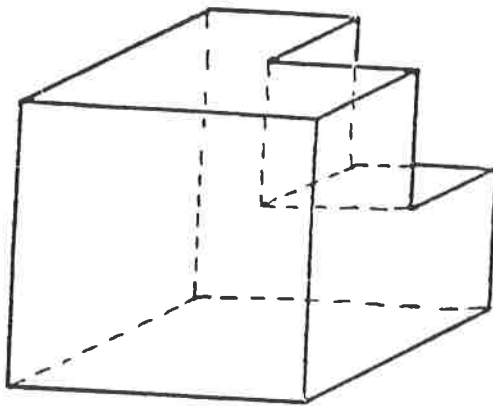


Figura 2. Una diferència entre els 2 cubs A-B

A la figura 2 es veu el sòlid resultant de fer la diferència A-B entre els 2 sòlids de la Figura 1.

La implementació d'aquest mètode s'ha dividit en 4 processos que s'expliquen detalladament en els següents apartats i que dit breument, fan el següent:

Procés 1: estudia tots els parells formats per una cara del sòlid A i una del B per veure si interfereixen o no i, si és que sí, calcula l'aresta o arestes de tall que se'n generen. Al final, cada cara tindrà

associada una llista d'arestes de tall provocades per la interferència amb cares de l'altre sòlid, i que serà nul·la en algunes cares que són aquelles que romanen intactes.

Procés 2: Aquest procés s'aplica a totes les cares que, a partir del procés 1, tenen associada una llista d'arestes no nul·la. I el que fa és obtenir totes les noves cares en que s'han subdividit cadascuna d'aquelles.

Per tant, amb aquests 2 processos ja tenim 'l'explosió de cares'. I a més sabem, per cadascuna d'elles, de quin sòlid provenen.

Procés 3: Determina per totes les cares que tenim actualment si són interiors o exteriors al sòlid de que no provenen.

Procés 4: Calcula a partir de la taula 1 quines son les cares que compleixen la propietat associada a l'operació booleana que es vol realitzar. I aquestes són les cares que formaran part del sòlid final.

Tal com es desprén del que s'acaba de dir, els processos 1, 2 i 3 només depenen dels sòlids i no de l'operació booleana, per tant faran el mateix independentment de l'operació booleana. L'únic que en depen és el procés 4.

Finalment, només cal fer dues puntualitzacions respecte als processos 2 i 4. Aquests dos processos seran més complexos del que pugui semblar en una primera lectura degut al fet que, en principi, els sòlids estaran formats per cares convexes i concaves i que, amés, podran tenir diversos forats (que són els que donen lloc a protuberàncies o concavitats als sòlids inicials). També es podrà donar el cas que, com a resultat de la interferència, desapareguin alguns d'aquests forats o se'n produeixin de nous. I, en segon lloc, en el procés 4, caldrà tenir en compte el sentit de la normal de les cares per tal que el model de fronteres del sòlid resultant sigui coherent.

### 3.- PROCES 1

#### 3.1 Introducció

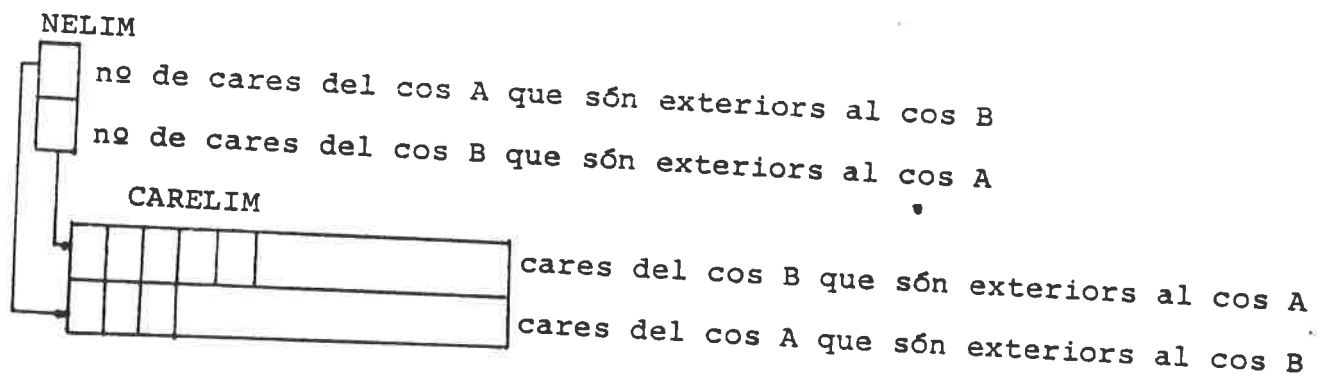
Aquest proces estudia les possibles interferències entre les cares del sòlid A i les del sòlid B i, quan ve al cas, determina les arestes de

tall que es generen.

Les dades d'entrada d'aquest procés són els models de fronteres de cadascun dels sòlids inicials. El model de fronteres consta de quatre estructures, cares, polígons, arestes i punts.

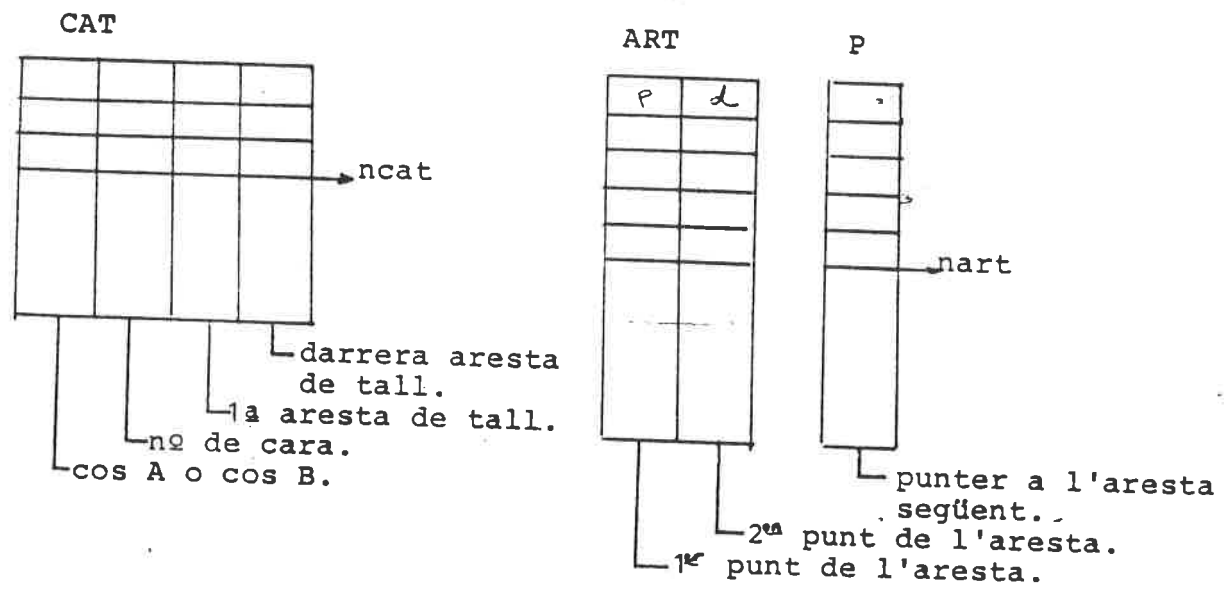
La sortida la podrien considerar com a dos blocs de dades:

Un primer bloc, resultant d'un procés en que per cada cara, es calcula la seva capsa mínima contenidora i aleshores es mira si aquesta i la capsa mínima contenidora de l'altre sòlid són o no disjunts. Si són disjunts vol dir dues coses: en primer lloc, que no cal aparellar aquella cara amb cap de les de l'altre sòlid per veure si interfereixen o no; i en segon lloc; ja sabem que aquella cara és exterior a l'altre sòlid (per tant aquest bloc de dades ens serà útil també en el procés 3)



Esquema 1. Estructura de dades a la sortida del procés 1. Bloc 1.

El segon bloc conté totes les cares que han patit la interferència i la llista d'arestes de tall (amb els nous punts que s'hauran produït) que tenen associada.



Esquema 2. Estructura de dades a la sortida del procés 1. Bloc 2.

### 3.2 Algorisme

En un primer pas de l'anàlisi descendent, l'algorisme del procés 1 seria el següent:

#### acció procés 1

calcul del valor de  $\xi$

filtratge inicial de cares

obtenció de les arestes de tall

#### fiacció

#### 3.2.1 Calcul del valor de $\xi$

El valor de epsilon el calcula en funció de les dimensions dels 2 objectes que interfereixen i val:

$$d_{\max} \cdot 10^{-8}$$

essent  $d_{\max}$  la dimensió màxima (llargada, amplada o alçada) de les capses mínimes contenidores dels 2 objectes).

#### 3.2.2 Filtratge inicial de les cares

Aquesta és l'acció que obté l'estructura de dades de sortida del procés 1, Bloc 1, es a dir, deixa en aquesta estructura aquelles cares que segur que no interferiran amb l'altre sòlid i que, a més, en són totalment exteriors.

#### 3.2.3 Obtenció de les arestes de tall

En aquesta acció s'han d'analitzar tots els possibles parells de cares formats per una cara del cos A i una del B. Si alguna de les dues cares figura com a resultat de l'acció anterior, es passa immediatament a estudiar el següent parell.

Quan es té un parell de cares que no han estat filtrades s'ha de veure si interfereixen o no i per això es fan els següents passos:

- 1) Càlcul de la recta  $\cap$  entre els plans que contenen a cadascuna de les 2 cares (Recta R de la fig. 3).
- 2) Calcul de les 2 arestes o els 2 conjunts d'arestes resultat de la intersecció entre la recta anterior i cadascuna de les cares. (Aresta P1-P3 i aresta P2-P4 de la figura 3)
- 3) Intersecció entre les 2 arestes o els 2 conjunts d'arestes anteriors

amb el que obtenim l'aresta o conjunt d'arestes definitiu. (Aresta P2-P3 de la figura 3)

- 4) Finalment, cal omplir l'estructura de dades de sortida d'aquest procés, Bloc2, amb aquestes noves arestes. Cal colocar les 2 cares suposant que no hi fossin ja, i a continuació cal colocar el conjunt d'arestes definitiu com noves arestes de tall de totes dues cares

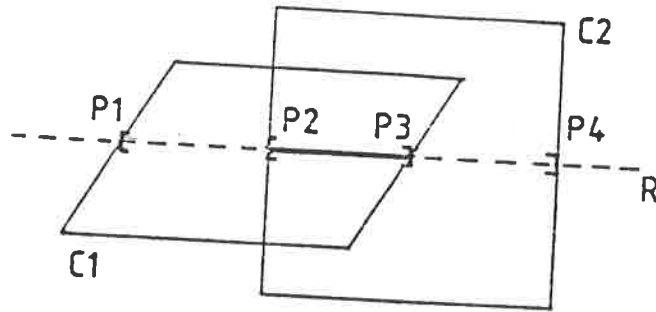


Figura 3. Obtenció de les arestes de tall

L'algorisme d'aquesta acció seria:

acció obtenció de les arestes de tall

per i=1 fins m fer; m:nº de cares del cos A

Si la cara i no ha estat filtrada llavors

càlcul dels coeficients del pla de la cara i

per j=1 fins n fer; n:nº de cares del cos B

Si la cara j no ha estat filtrada llavors

càlcul dels coeficients del pla de la cara j

" de la recta intersecció

" de la intersecció entre la recta i la cara i:arestes-

" " " " " " " " " " " j:arestes-

" " " " " els conjunts d'arestes-i i

arestes-j

col.locació dels resultats a l'estructura de dades de

sortida

fisi

fiper

fisi

fiper

fiacció

En aquest algorisme es resolen problemes que inicialment són 3D però que realment s'han pogut resoldre en 2D. El problema de la intersecció

entre una recta i una cara a l'espai, s'ha passat al pla (per tal de simplificar-ho), projectant la cara i la recta a un dels 3 plans coordenats. El pla a projectar, s'ha triat en funció de la cara i s'ha escollit aquell que és perpendicular a la component més gran del vector normal de la cara. D'aquesta manera s'obtenen les màximes projeccions i s'eviten les projeccions nul·les o quasi nul·les. La intersecció entre una recta i una cara al pla és un problema àmpliament resolt [6] i la intersecció entre 2 conjunts d'arestes és, en el fons, un problema d'ordenació.

### 3.3 Comentaris

El procés 1 és el més crític, es a dir, el més costós, degut a que ha de fer la comparació entre tots els possibles parells de cares i el càlcul de la possible intersecció entre elles. Aquest procés té complexitat  $n^2$ .

Aquest procés es podria agilitzar fent un altre filtratge inicial de les cares (verure 3.2.1) que consistiria en que, abans de calcular la intersecció entre les 2 cares, veure si les capsos mínimes contenidores de cadascuna d'elles, són disjunctes o no. Si son disjunctes, ja es podria passar a analitzar el següent parell ja que aquestes 2 segur que no interseccionen.

Finalment, cal comentar aquí l'avantatge que ofereix l'esmentat model d'octrees (1.1) en la realització d'operacions booleanes. Bàsicament, el que s'estalvia es la feina que dur a terme aquest procés que, com s'ha dit, és la més crítica. En el model d'octrees tenim els sòlids descomposats en nodes blancs o negres i només cal anar comparant els nodes d'un sòlid i altre que coincideixen a l'espai, per tal d'obtenir el node corresponent del sòlid resultant. Per tant no cal dur a terme el procés de comparació cara-cara.

## 4.- PROCES 2

### 4.1 Introducció

El procés 2 tracta totes aquelles cares que tenen arestes de tall a la sortida del procés 1. Es un procés repetitiu per totes aquelles cares i el que fa és obtenir totes les noves cares resultat de la interferència.

El procés distingeix clarament entre les arestes pròpies de la cara que són les que ha tingut sempre (incloses les dels seus possibles polígons) i les arestes de tall, que són les que ha calculat el procés. Una cara pot quedar convertida en una o més cares cadascuna de les quals podra tenir polígons forats i caldra distingir entre els polígons vells que són els que provenen del cos inicial i els polígons nous que s'hagin pogut generar degut a la interferència. Finalment cal fer notar que, de la mateixa manera que poden apareixer polígons nous, en poden desaparèixer de vells.

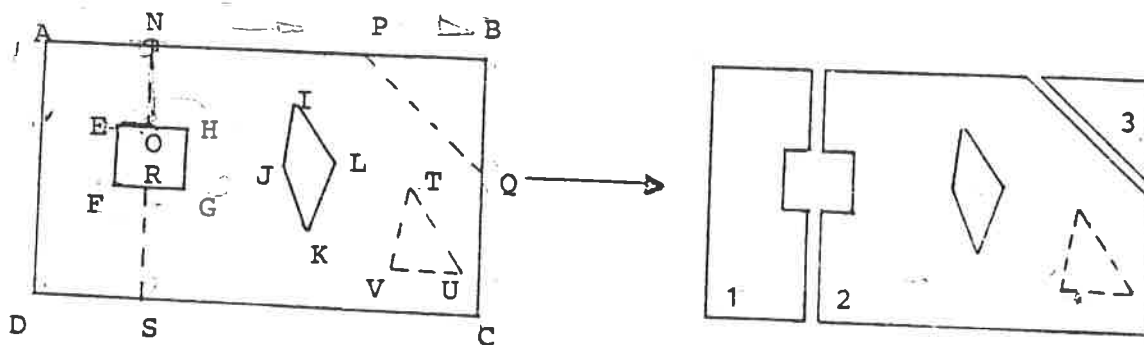


Fig.4. 'Explosió d'una cara'. — arestes pròpies.  
 - - - - - " de tall.

#### 4.2 Mètode i algorisme.

En primer lloc, cal deixar clars uns conceptes. Cada cara donarà lloc a una o més cares formades pel seu contorn extern i els possibles polígons vells i nous. Tots aquests contorns externs estaran formats per arestes o trossos d'arestes pròpies i per arestes de tall (aquestes sempre senceres). Les arestes de tall formaran part sempre de 2 contorns externs (El cas de les arestes de tall que donen lloc a polígons nous també compleixen aquesta propietat, però això es veurà en més detall en el procés 4). Les arestes pròpies només formen part d'un contorn extern. Com es pot veure a la figura 4, algunes arestes pròpies hauran quedat subdividides en 2 o més arestes que caldrà calcular donant lloc a una "llista extesa d'arestes pròpies". Finalment, l'aparició de polígons nous es detectarà mitjançant el fet que, un cop s'han obtingut tots els possibles contorns externs, hi ha arestes de tall que no han intervingut en cap d'ells; tots els possibles bucles a que donin lloc aquestes arestes de tall seran polígons nous. Evidentment, si la abans esmentada llista extesa d'arestes pròpies coincideix exactament amb l'antiga, es a dir, que cap aresta pròpia ha estat subdividida, totes les arestes de tall formaran polígons nous i l'únic contorn extern que hi haurà serà el de la cara inicial.



Amb aquestes consideracions veiem l'algorisme del procés 2

acció procés 2

per  $i=1$  fins  $n_{cat}$  fer ; per totes les cares que tenen arestes de  
bucles(i) tal

fiper

fiacció

acció bucles(i)

colocar les arestes pròpies i de tall de la cara en 2 estructures de dades.

obtenir la 'llista extesa' d'arestes pròpies

Si llista extesa = llista antiga llavors

només donarà lloc a una cara, de la qual ja coneixem el contorn extern i els polígons vells

obtenir els polígons que s'han generat

assignar aquests bucles com a polígons nous de la cara

altrament

obtenir tots els possibles contorns externs

si queden arestes pròpies sense utilitzar llavors

obtenir els polígons vells que no han desaparegut

fisi

si queden arestes de tall sense utilitzar llavors

obtenir els polígons nous que s'han generat

fisi

si hi ha polígons nous o vells llavors

redistribuir-los entre els contorns externs generats

fisi

fifi

fiacció

A continuació, més que anar baixant en l'anàlisi descendent de l'algorisme, el que es farà és explicar com s'ha dut a terme cadascuna de les accions.

#### 4.2.1 Obtenció de la llista extesa.

Per això cal recórrer totes les arestes pròpies i veure tots els possibles punts extrems de les arestes de tall que estan a sobre d'elles. A continuació cal ordenar els punts, obtenint així la nova aresta subdividida. A la figura 4, l'aresta pròpia superior conté 2 punts extrems de 2 arestes de tall i, per tant, queda convertida en 3 are-

#### 4.2.2 Obtenció de polígons vells i nous

Tant en un cas com en l'altre és el típic problema que a partir d'un conjunt d'arestes va formant tots els possibles bucles anant-les enllaçant. A la figura 4 es pot veure que ha desaparegut un polígon vell (EFGH), l'altre s'ha mantingut (IJKL) i n'ha aparegut un de nou (TUV).

#### 4.2.3 Obtenció dels contorns externs

Aquesta es l'acció més complexa d'aquest procés. En realitat també es tracta de fer bucles, però utilitzant a la vegada arestes pròpies i arestes de tall, seguint unes determinades normes. El mètode comença sempre amb una aresta pròpia, però al cercar l'aresta següent a enllaçar, com que potser una aresta pròpia o una de tall, dona preferència a la de tall, de forma que el nou contorn es va movent per les noves fronteres (arestes de tall) que ha provocat la interferència entre els 2 cossos.

A la figura 4, per formar el contorn 1, per exemple, s'ha començat amb l'aresta AN, a continuació ha calgut triar entre la NP o la NC i s'ha escollit la NO perquè és aresta de tall.

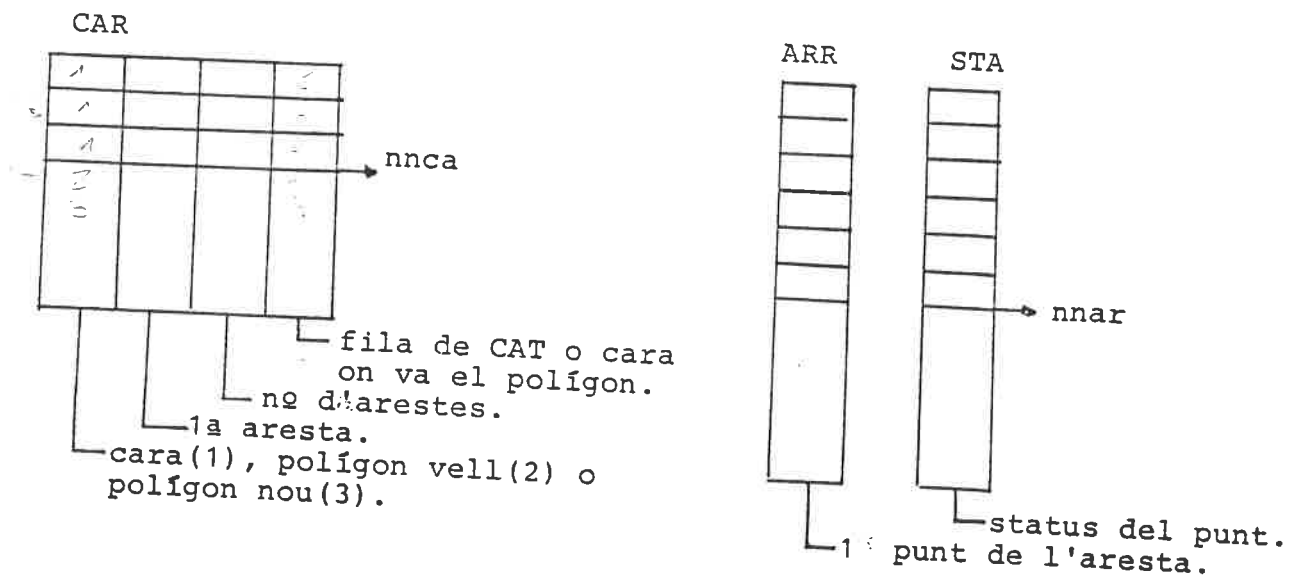
En aquesta acció, cada cop que s'utilitza una aresta s'ha d'indicar ja que, com s'ha vist, les arestes pròpies només intervindran en un sol contorn i les de tall només en 2. Així, entre altres coses permetrà la posterior obtenció de polígons vells i nous.

#### 4.2.4 Redistribució dels polígons

Se soluciona utilitzant el conegut mètode de veure si un punt és interior a un polígon. S'agafa un punt de cada polígon vell o nou i es mira a quin dels nous contorns externs és interior.

### 4.3 Estructura de dades de la sortida.

L'estructura de dades a la sortida és un conjunt de bucles: contorns externs, polígons nous i polígons vells amb totes les arestes que els constitueixen. A més també s'indica, pels polígons, a quin contorn extern pertanyen.



Esquema 3. Estructura de dades a la sortida del procés 2.

Com es veu a l'esquema, les arestes estan posades en ordre i cadascuna ve representada pel seu primer punt, en el vector ARR. Associat a aquest vector n'hi ha un altre, STA en el que es dóna el 'status' dels primers punts de cada aresta. Aquests codis poden tenir els següents valors

- 1: punt 'significatiu' (punt antic, que ja formava part del model de fronteres del cos inicial)
- 2: punt inicial d'una aresta de tall
- 0: ni 1 ni 2.

Aquest valor es pot anar obtenint fàcilment a mida que s'obtenen els contorns externs i polígons. I es farà servir en el procés 3 on es veurà la seva utilitat.

## 5.- PROCES 3

### 5.1 Introducció

En aquest procés es determina per cadascuna de les noves cares (contorns externs) generades en el procés 2 i també per totes aquelles cares que no han estat interferides, si són interiors o exteriors a l'altre cos.

### 5.2 Mètode i algorisme

Per saber si una cara és interior o exterior a l'altre cos, cal agafar un punt de la cara i aplicar-hi l'algorisme de punt exterior a un sòlid. Aquest punt no pot ser qualsevol sinó que ha de ser un

punt que no estigui a la frontera de l'altre cos, és a dir, que no pertanyi a una aresta de tall de les obtingudes al procés 1. Aquests punts són els que tenen status 1 (punts significatius) en el procés 2.

De totes maneres, no caldrà aplicar l'algorisme per cada cara ja que la condició d'interior o exterior d'una cara, es pot transmetre a les altres cares del mateix cos per relacions de veinatge, seguint els següents criteris. Un cop se sap la condició d'una cara (p.ex. interior) aquesta es la condició de tots els seus punts significatius i, per tant, tota altra cara que comparteixi punts significatius amb aquella té la mateixa condició (interior) i així, via punts significatius, podrem coneixer la condició de moltes cares. Hi haurà un moment, però, en que el procés s'aturarà, i aleshores caldrà utilitzar la informació de les arestes de tall. Com s'ha dit en el procés 2, una aresta de tall sempre pertany a 2 de les noves cares i degut a que aquesta aresta de tall és la que ha provocat precisament una escisió, podem afirmar que les 2 noves cares que comparteixen una aresta de tall tenen condició contrària. Per tant, també podem transmetre informació via arestes de tall. Reconeixem que una aresta és de tall perquè el seu punt inicial té status 2 tal com s'ha indicat en el procés 2.

Vist des d'un altre punt de vista, podem considerar que cadascun dels 2 cosos inicials s'ha convertit en  $n$  cosos més petits, degut a la interferència i que es toquen entre sí. Totes les cares de cadascun d'aquests  $n$  cosos tenen la mateixa condició. Aleshores es pot afirmar que el pas d'informació entre les cares d'un d'aquests  $n$  cosos es pot fer totalment via punts significatius, però quan s'ha de passar d'un a l'altre, cal fer-ho via arestes de tall. A la fig 5, es veu els 2 cosos en que s'ha trencat el cos A de la fig. 1. Si calculem per ex. que el punt D és exterior al cos B, podem saber immediatament, via punts significatius que totes les cares del cos 1 són exteriors, però no podem transmetre la informació al cos 2 sino mitjançant les arestes de tall que són totes les dobles, i així sabem que totes les cares del cos 2 seran de condició contrària i per tant interiors al cos B.

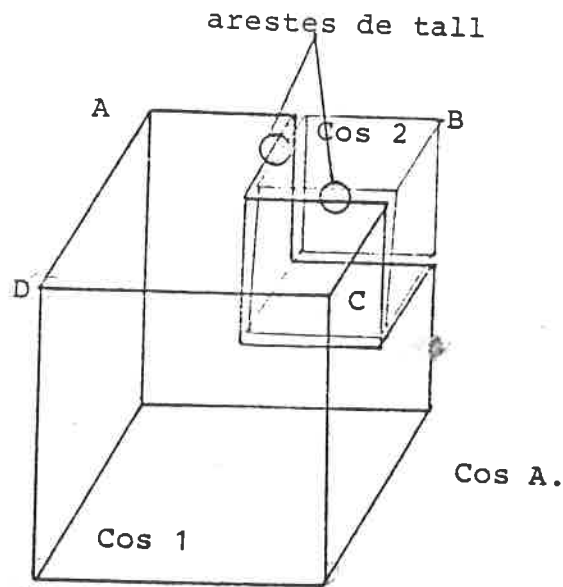


Fig. 5. Pas d'informació al procés 3.

Per tant, seguint aquest mètode, només caldrà aplicar l'algorisme de punt interior a un sòlid, com a màxim a un sol punt de cadascun dels 2 cossos inicials.

I si es diu com a màxim és perquè hi haurà casos en que no caldrà aplicar-lo cap cop. Com s'ha vist, en el procés 1 es fa un filtratge inicial d'unes determinades cares que són exteriors. Doncs bé, només que en aquell procés haguem filtrat una cara de cada cos ja tenim informació per arrencar el procés 3 sense aplicar cap cop algorisme de punt interior a un sòlid.

Amb aquestes consideracions ja podem escriure la primera fase de l'algorisme, que bàsicament el que fa és anar cercant cares de les que pugui conèixer la condició (interior o exterior).

Inicialitzacions

per  $i=1$  fins 2 fer ; per cadascun dels 2 cossos inicials

construeix una llista de cares no tallades del cos  $i$

si el procés 1 ha filtrat algunes cares llavors

actualitzar-les ; totes són exteriors

altrament

cercar una cara que tingui algun punt significatiu

mirar si aquest punt és interior o exterior a l'altre cos

actualitzar aquesta cara

fisi

mentre quedin cares per actualitzar del cos  $i$  fer  
 cercar la primera de la que es pugui conèixer la condició  
 actualitzar-la

fimentre

fiper

### 5.2.1 Llista de cares no tallades

Es construirà senzillament agafant, de totes les cares inicials d'un cos, totes menys les que el procés 1 dóna com a sortida de cares amb arestes de tall.

### 5.2.2 Actualitzar una cara.

Aquesta acció té dues missions, per un costat deixa constància, en l'estructura de dades que es detallarà més endavant, de la condició que s'acaba de veure que té aquella cara. I, a més, també es guarda la informació de tots els seus punts significatius, associats amb la condició que sabem que tenen i també de totes les arestes de tall. Això és necessari per tal de poder passar la informació d'aquesta cara a les altres. Els punts significatius es van guardant tots i mai no s'esborren perquè poden pertanyer a moltes cares, en canvi les arestes de tall només pertanyen a dues cares, per tant, quan es troben per segona vegada, se'n consulta la condició i s'esborren.

### 5.2.3 Cercar una cara de la que en puguem conèixer la condició.

Aquesta acció recorre totes aquelles cares del cos i que encara no estan actualitzades, fins que en troba alguna que té un punt significatiu del que se'n coneix la condició. Alehores la condició de la cara és la mateixa que la d'aquest punt.

Si no se'n troba cap, es fa el mateix cercant la primera cara que tingui una aresta de tall de condició coneguda. En aquest cas, la condició de la cara serà la contrària a la de l'aresta.

I es pot demostrar que sempre es trobarà una cara amb un punt significatiu o una aresta de tall coneguts.

### 5.2.4 Cercar alguna cara amb un punt significatiu

En el cas que el procés 1 no hagi pogut filtrar cap cara, haurem d'engegar aquest procés aplicant l'algorisme de punt interior a un sòlid, però aquest punt com ja s'ha vist ha de ser significatiu (sino cauria a la frontera de l'altre cos). Tot i que poden haver-hi noves cares que no tinguin cap punt significatiu -a la fig. 6 la cara 2 no en té cap-, el més probable és que la primera cara que con-

sultem en tingui algun.

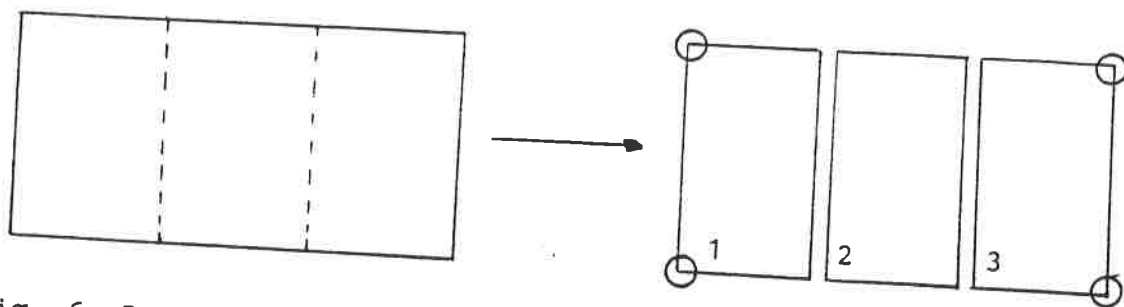


Fig. 6. La cara 2 no té cap punt significatiu. (○)

### 5.2.5 Punt interior a un sòlid.

Aquest problema es resol de la següent manera. Es crea un pla horitzontal que passi pel punt a determinar. A continuació s'obtenen totes les arestes provinents de la intersecció entre aquest pla i totes les cares i polígons del sòlid. D'aquesta forma aconseguim passar un problema de l'espai al pla (al pla horitzontal esmentat). Un cop ens situem al pla, cal reorganitzar totes les arestes obtingudes formant un o més bucles, és a dir, una figura plena de diversos polígons. I a partir d'aquí, ja es pot aplicar el conegut algorisme de punt interior a un polígon.

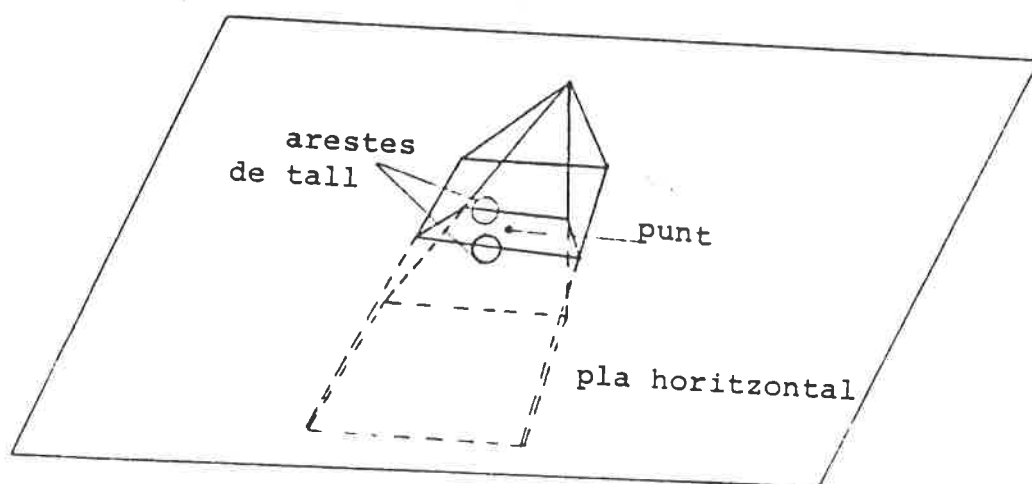
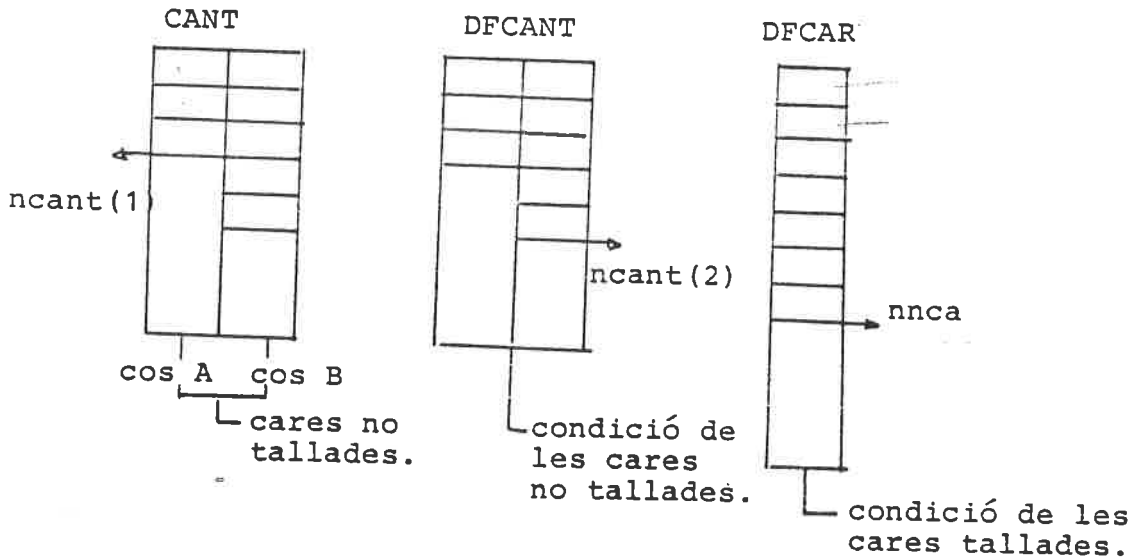


Fig. 7. Punt interior a un sòlid.

### 5.3 Estructura de dades a la sortida

El que calcula el procés 3 ho deixa en 3 estructures que són les

que es veuen a l'esquema



Esquema 4. Estructura de dades a la sortida del procés 3.

la matriu CANT és la que conté, a cada columna, la llista de cares no tallades de cada cos. La matriu DFCANT tindrà la mateixa dimensió real que la CANT i conté la condició de les cares referenciades a CANT. Finalment el vector DFCAR té el mateix nº de files que la matriu CAR sortida del procés 2 -que era la que contenia tots els contorns i polígons resultat de les cares amb arestes de tall- i en els llocs corresponents a les cares o contorns externs hi ha indicada també la condició d'aquella cara.

## 6. PROCES 4

### 6.1 Introducció

Amb el procés 4 finalitza la realització de l'operació booleana que es volia. Com ja s'ha dit al capítol 2, en els 3 processos anteriors no es necessitava saber quina era l'operació booleana ja que s'havien de fer exactament igual per a qualssevol d'elles. I és el procés 4 el que, sabent l'operació booleana que es vol fer i basant-se en la taula 1, determina totes les cares que aniran al cos final. De totes maneres, aquest procés no és tant simple com això ja que ha de resoldre, a més, dos problemes: un relacionat amb el sentit del vector normal de les cares i el fet que la seqüència d'arestes en tots els polígons d'una cara ha d'estar d'acord amb el sentit de la normal especificat, i l'altre té que veure amb els polígons nous que s'han generat.



### 6.1.1 Sentit del vector normal

En aquest procés cal veure quin sentit de la normal han de tenir les cares i a continuació comprovar si tots els polígons que hi pertanyen estan d'acord amb aquest sentit -mirant l'ordre en que estan les seves arestes frontera-.

Si es mira la figura 1, es pot observar el següent:

- Totes les noves cares produïdes son part o la totalitat de cares antigues i, per tant, si més no, tenen la mateixa direcció del vector normal de les cares antigues de que provenen. Per tant, no s'hauran de calcular noves components de nous vectors normals si-nó que només caldrà veure si el vector normal canvia o no de sentit.
- Totes les noves cares que van a la unió i a la intersecció, mantenen el sentit del vector normal de les cares de que provenien. En canvi, les cares que van a les diferències compleixen la següent propietat: les cares del sòlid al qual restem i que són exteriors a l'altre, també van al cos final amb el mateix sentit del vector normal de les cares de que provenien, pero les cares del cos que restem, que són interiors a l'altra, no. Podriem dir que, per aquestes cares, el volum al que fan de frontera, ha canviat de cantó.

Un cop sabem el sentit del vector normal que ha de tenir una cara, hem de comprovar si tots els polígons que hi pertanyen hi estan d'acord. El polígon que determina el contorn de la cara i els polígons vells d'aquella cara seguiran el mateix procés que el propi sentit de la cara, és a dir, romandran intactes si el vector normal no canvia i hauran de canviar l'ordre de les seves arestes en el cos esmentat de les cares que van a una diferència i que són del cos que restem. Això és degut a que els polígons contorns, o bé coincideixen amb el contorn d'una cara antiga -cara no tallada- o bé en mantenen l'ordre, ja que l'algorisme del procés 2 així ho fa (el contorn es comença a fer sempre amb una aresta propia o antiga i aquesta determina l'ordre de les arestes segons el de la cara antiga). Els polígons vells que es mantenen, sempre romanen intactes.

Ara bé, els polígons nous són contorns nous que s'han format amb arestes de tall. I per fer un d'aquests polígons, s'ha agafat la primera d'aquestes noves arestes que s'ha trobat, que, a més, va en un sentit totalment aleatori. Per tant, per aquests polígons, caldrà cal-

cular el sentit del vector normal que determinen les seves arestes inicialment, veure si coincideix amb el que ha de tenir, i si no és així, canviar l'ordre de les arestes.

#### 6.1.2 Polígons nous.

La producció de polígons nous pot donar lloc a una complicació del problema tot i que s'han de tenir en compte tots els casos possibles, la complicació del problema va lligada amb el n°de polígons nous que es produeixin i per tant, com més complicat és el cas, més improvable és que es produeixi.

En primer lloc, cal veure que els polígons nous i els vells es comporten de forma molt diferent. D'entrada cal veure que els polígons nous podran ser forats d'una cara, podran ser cares i també podran ser forats d'un altre polígon nou que sigui cara i tot això s'haurà de determinar. Els polígons vells només poden ser forats, però ho poden ser d'una cara o d'un polígon nou que sigui cara i aleshores realment seran forats, o bé ho poden ser d'un polígon nou que sigui forat i en aquest cas no seran considerats (un forat en un forat no té sentit).

Degut a la improbabilitat d'aquests casos, és difícil trobar i representar un exemple. De totes maneres en el següent es poden veure algunes de les propietats esmentades i se'n poden imaginar les altres.

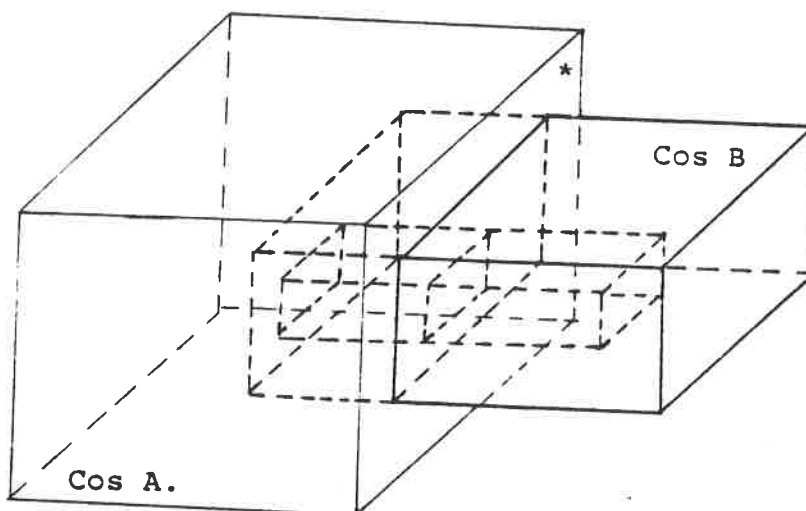


Fig. 8. Proces 4. Polígons nous.

A la figura 8, es veuen 2 cossos, el cos A és simplement un paral.lelepíped i el cos B un altre amb un forat també paral.lelepípedic que té a la cara marcada amb una creu. Estudiarem que li passa a la cara del cos A marcada amb un \*

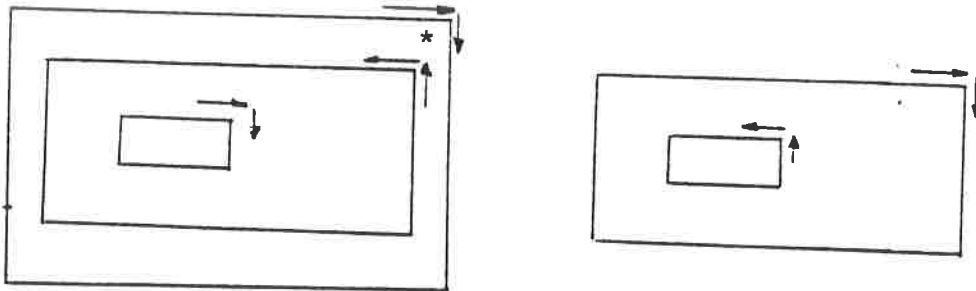


Fig. 9. Cara (\*) del cos A.

Aquesta es una cara amb 2 polígons nous. I ara es veura que és el que anirà a cada operació booleana d'aquesta cara:

A la unió, p1 és el contorn d'una cara, p2 és un forat d'aquesta cara i p3 és una altra cara.

A la intersecció, p1 no hi surt, p2 és una cara i p3 un forat d'aquesta cara.

A la diferència A-B podem dir el mateix que a la unió i a la diferència B-A, el mateix que a la intersecció.

A la figura 10 es veuen els 4 cossos resultants de les 4 operacions booleanes.

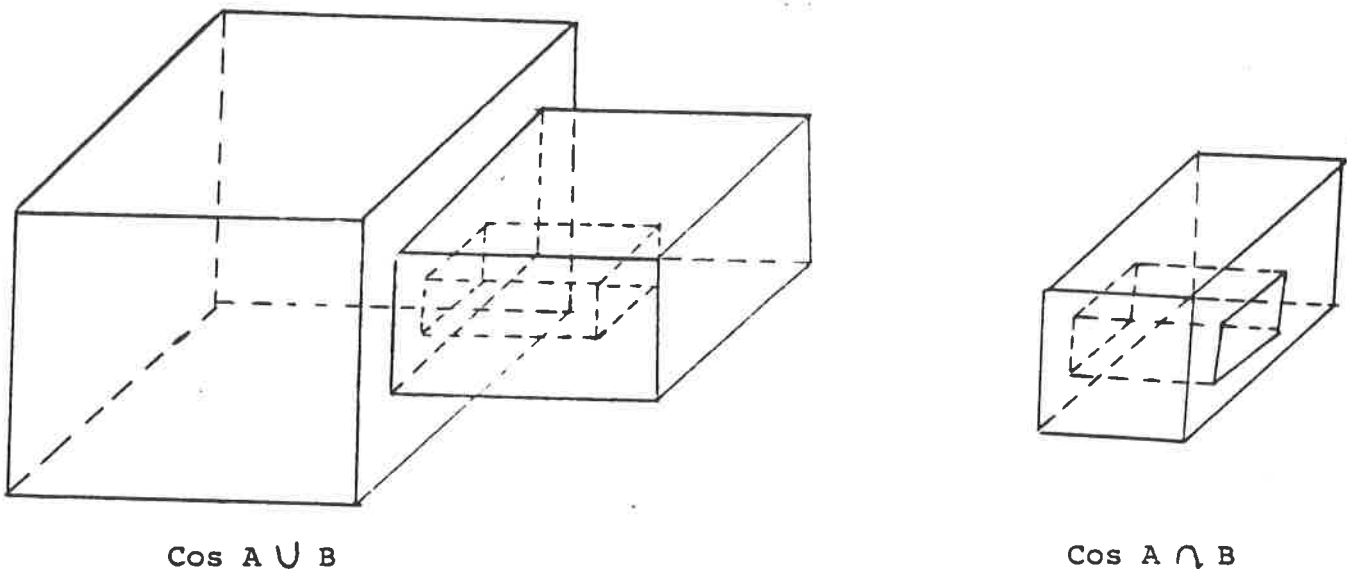
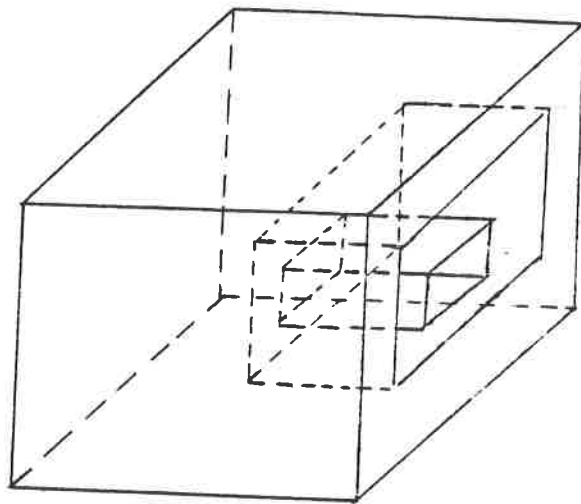
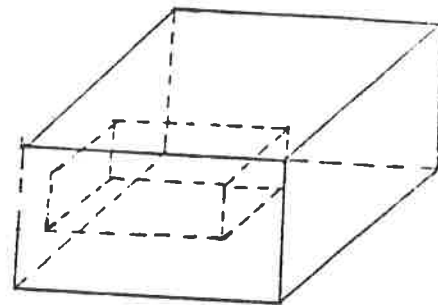


Figura 10. Operacions Booleanes dels objectes de la fig. 8.



Cos A-B



Cos B-A

fig. 10. Operacions Booleanes dels objectes de la fig. 8.

Si ara ens imaginéssim que el cos A tenia un forat a la cara amb un asterisc (polígon vell) i aquest forat quedava interior a  $p_2$  però exterior a  $p_3$ , aleshores aquest forat passa a pertanyer a  $p_2$  i només apareixerà quan  $p_2$  és cara ja que quan  $p_2$  es un forat no té cap sentit ja que el forat polígon vell queda totalment anul·lat pel forat polígon nou que és més gran.

Anem a veure ara com es farà l'estudi d'aquestes cares més especials, que seran totes aquelles que tinguin polígons nous.

D'entrada caldrà veure quins polígons nous i vells van al cos final i com hi van (com a cares o forats). Per aixó caldrà estudiar la distribució dels polígons dins de la cara, és a dir, caldrà saber per cada polígon nou quin és el seu més immediat contenidor i quins són els seus immediats continguts. Per cadascún dels polígons vells només caldrà saber quin és el seu més immediat contenidor.

Començant pels polígons nous, si són disjunts entre sí, llavors és un cas simple en que la cara mare donarà lloc a: la cara amb els polígons com a forats i tantes cares com polígons nous hi havia. En aquest cas, qualsevol entitat que va com a cara haurà de tenir el mateix sentit del vector normal que la cara mare i si va com a forat, el sentit del vector normal serà el contrari. Veure figura 12

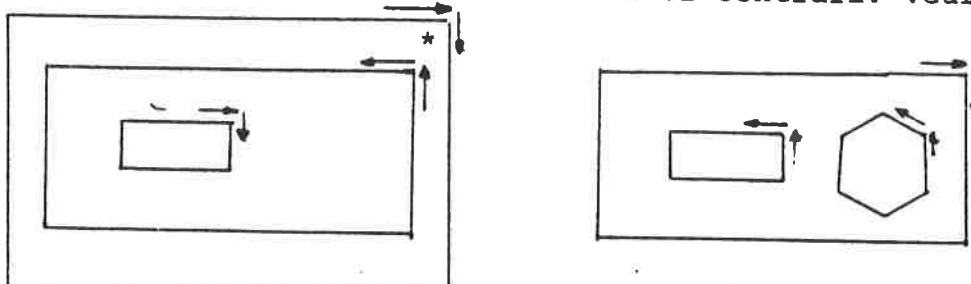


Fig. 11. Cara del cos A amb forat vell.

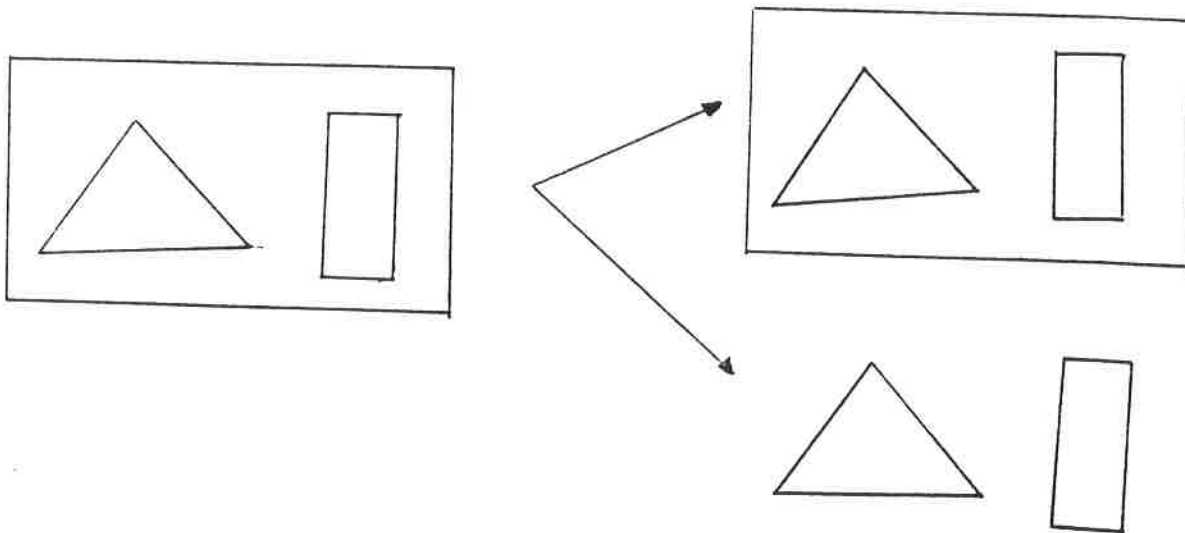


Fig. 12. Cara amb forats nous disjunts.

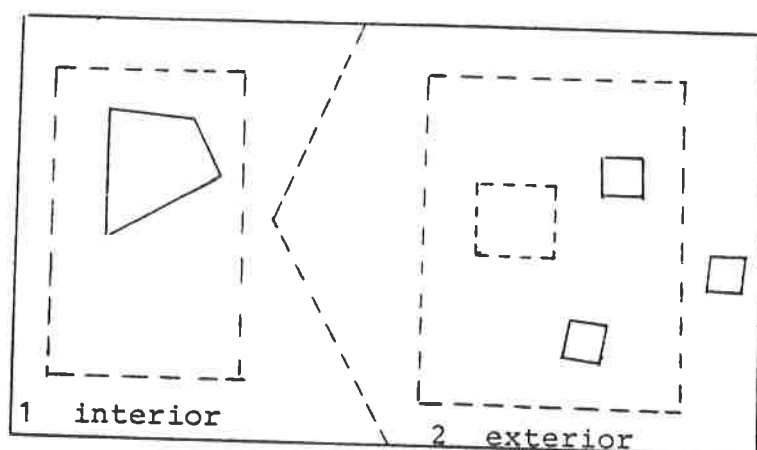
Si els polígons nous no són disjunts entre sí, aleshores el cas és més complicat i caldrà trobar les relacions d'inclusió entre ells. tot polígon nou serà considerat forat del seu més immediat contenidor (sigui un altre polígon nou o la cara mare) i a la vegada serà considerat cara que tindrà com a forats aquells polígons nous i vells més immediatament continguts per ell. El paper que jugarà (cara o forat) quan sigui assignat al cos final, dependrà de l'operació booleana demanada.

A la fig. 13 es veu un cas una mica complicat. Una cara s'ha convertit en dues cadascuna de les quals té diversos polígons nous i vells (procés 2), sabem que la cara 1 és interior i la 2 exterior (procés 3). Dones bé, el procés 4 troba que hi ha més de 2 cares que són les que s'indiquen a sota.

Finalment, només cal tornar a remarcar que si bé aquests casos es poden presentar algun cop, en la majoria de les cares que surten del procés 2 hi hauran 0 o 1 polígons nous.

## 6.2 Mètode i algorisme

Per dur a terme aquest procés es van estudiant totes les cares dels 2 cossos per veure si van o no al cos final. Es comença per les cares que han quedat intactes i després es passa a les tallades. I en aquestes, si hi ha més d'un polígon nou, es fa l'estudi explicat a l'apartat 5.1.2.



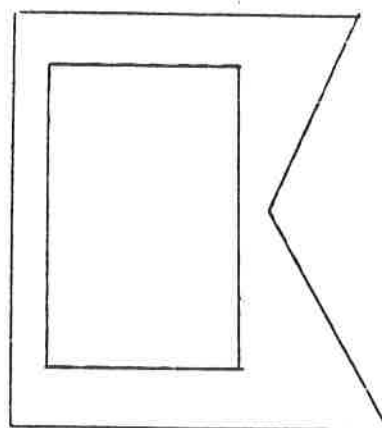
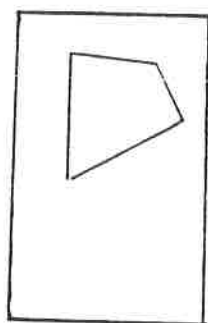
a)

cares externes

cares internes

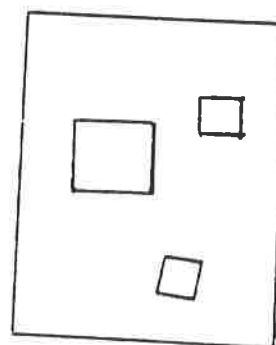
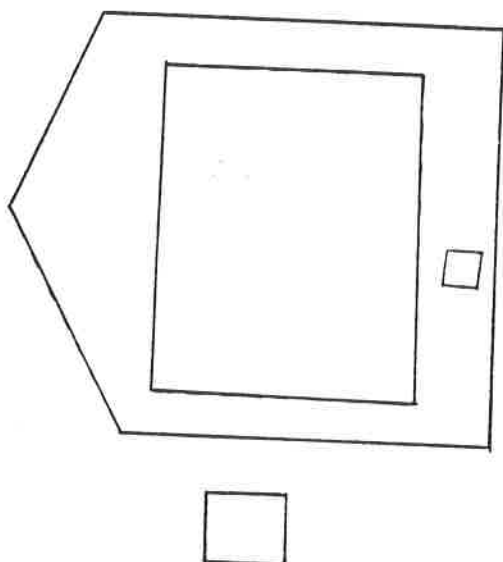
Subcara

1



Subcara

2



b)

Fig. 13. Cara amb forats nous no disjunts. a) cara inicial amb arestes pròpies (—) i arestes de tall (---). b) 'Explosió' de cares final, calculada pel procés 4.

Veiem l'algorisme:

acció procés 4

per i=1 fins 2 fer ; pels 2 cossos

passar les cares no tallades

" " " tallades

fiper

fiacció

acció passar cares no tallades

per j=1 fins nca t(i) fer

Si la cara va al cos final llavors

passar la informació de la cara (cara, polígon, aresta i punts)

fisi

fiper

fiacció

acció passar cares tallades

per j=1 fins nca fer

Si la cara va al cos final llavors

passar la informació de la cara (cara, polígon, arestes i punts)

fisi

comptar els polígons nous de la cara, npn

Si npn > 1 llavors

Si npn > 1 llavors

Veure com queden distribuïts (relacions d'inclusió entre ell

fisi

redistribuir els polígons vells en els nous

fisi

Si la cara va al cos final llavors

passar els polígons vells i nous immediats de la cara

fisi

passar adequadament la resta de polígons nous

fiper

fiacció

### 6.2.1 Condició que una cara vagi al cos final

La condició que una cara vagi al cos final és la següent:

i=n° del cos

codi= operació booleana

1: Unió

2: Intersecció

3: Diferència A-B

4: " B-A

condició  $\equiv$  (codi=1 o codi=3 i i=1 o codi=4 i i=2) i cara exterior o  
 (codi=2 o codi=3 i i=2 o codi=4 i i=1) i cara exterior

Es a dir, si una cara és exterior hi anirà sempre que fem una unió o si fem A-B i la cara és de A o si fem B-A i la cara és de B i si és interior hi anirà sempre que fem una intersecció o si fem A-B i la cara és de B o si fem B-A i la cara és de A.

#### 6.2.2 Passar informació de cares i polígons vells

Aquestes accions es limiten a omplir les matrius de cares, polígons, aristes i punts del cos de sortida amb la informació corresponent.

#### 6.2.3 Distribució de polígons nous i vells

Aquestes accions han de determinar exactament per cada polígon quin és el seu immediat contenidor. Sabent això, es fa fàcil obtenir, quan sigui necessari, tots els polígons nous i vells que són més immediatament continguts per la cara o per un polígon nou.

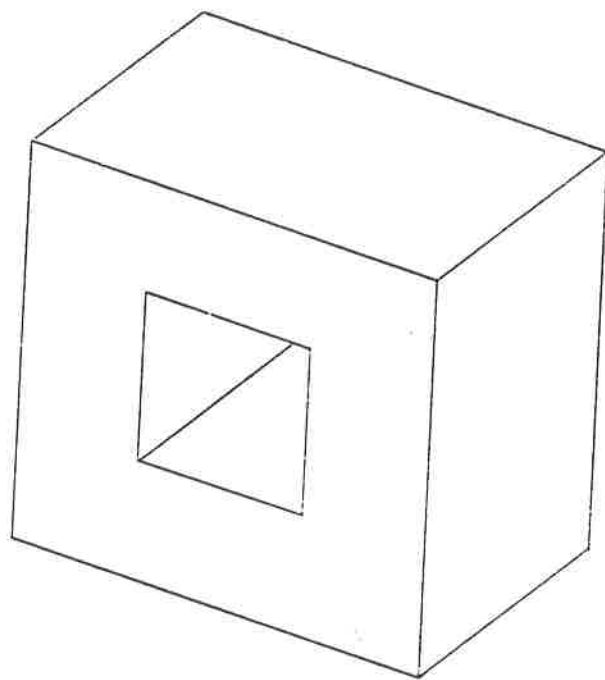
#### 6.2.4 Passar adequadament la resta de polígons nous.

De fet, els polígons nous sempre van al cos final sigui quina sigui l'operació booleana que es fa. Però com que hi poden anar com a cara o com a forat, això és el que cal determinar. Abans de fer aquesta acció, hem distribuït tots els polígons de forma que sabem exactament com estan. Per veure com es determina la forma anirem al cos final, cal imaginar-se que els polígons nous juntament amb la cara formen unes corbes (poligonals) de nivell. I aquestes corbes van alternativament al cos final com a cara, forat, cara, forat, etc... És a dir, si la cara hi va (i només hi pot anar com a cara) els polígons nous més immediats continguts hi van com a forats, els més immediats continguts d'aquests hi van com a cares i així successivament. Si la cara no hi va, tots els polígons juguen el paper contrari. La cara 2 de la figura 13 il·lustra un cas amb algunes de les característiques que s'acaben d'exposar.

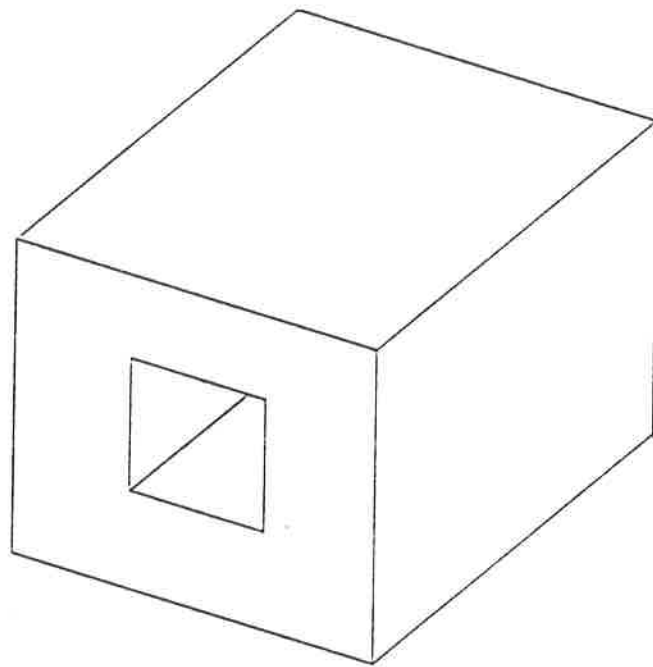


EXAMPLE 1

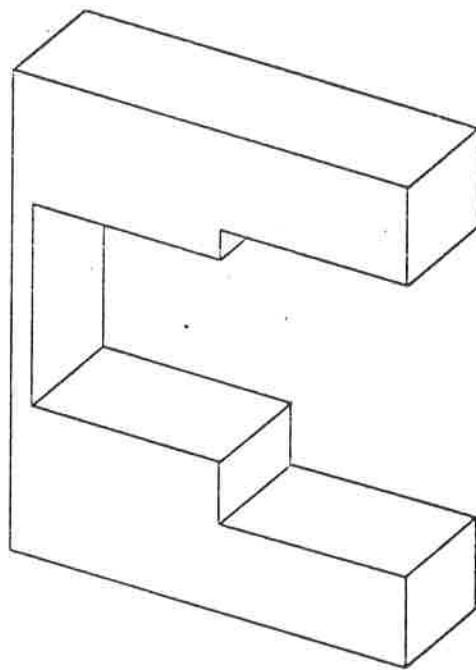
=====



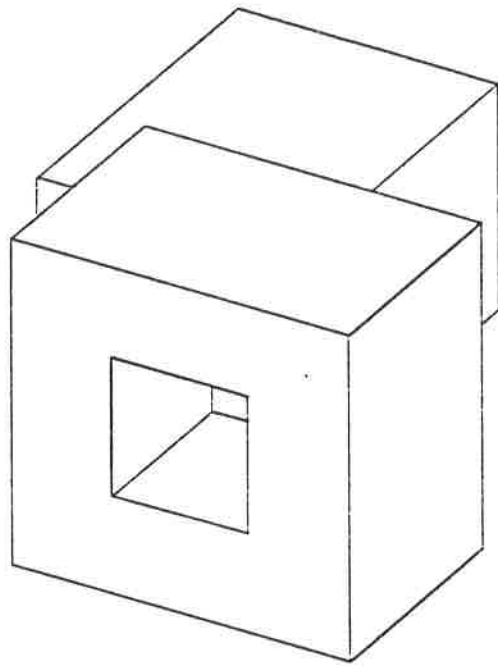
Cos B.



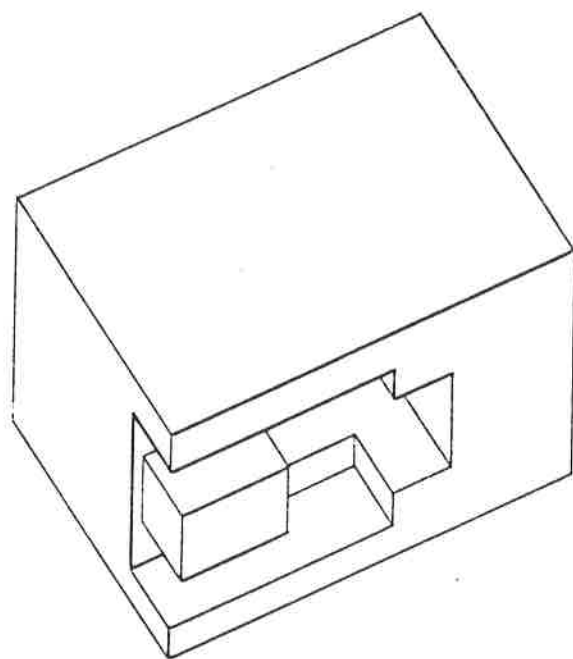
COS A.



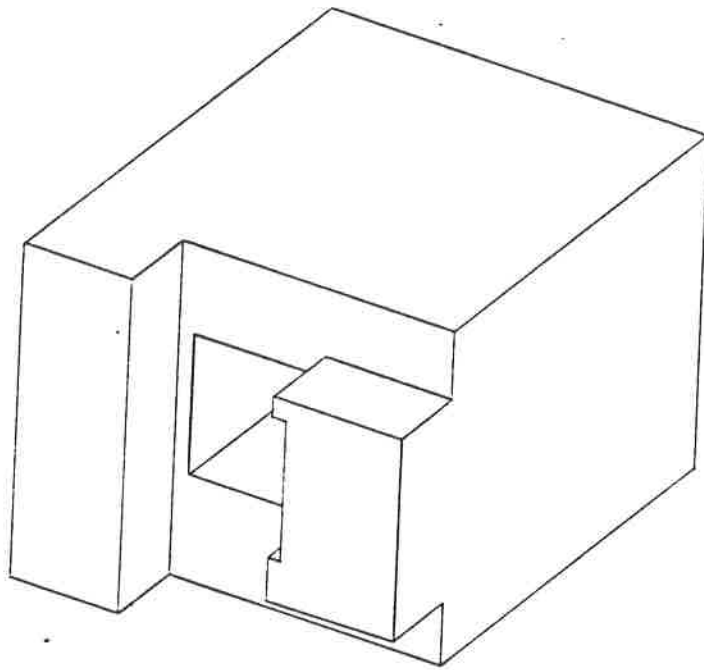
Cos  $A \cap B$ .



Cos A  $\cup$  B.



Cos A-B (Vist per darrera).

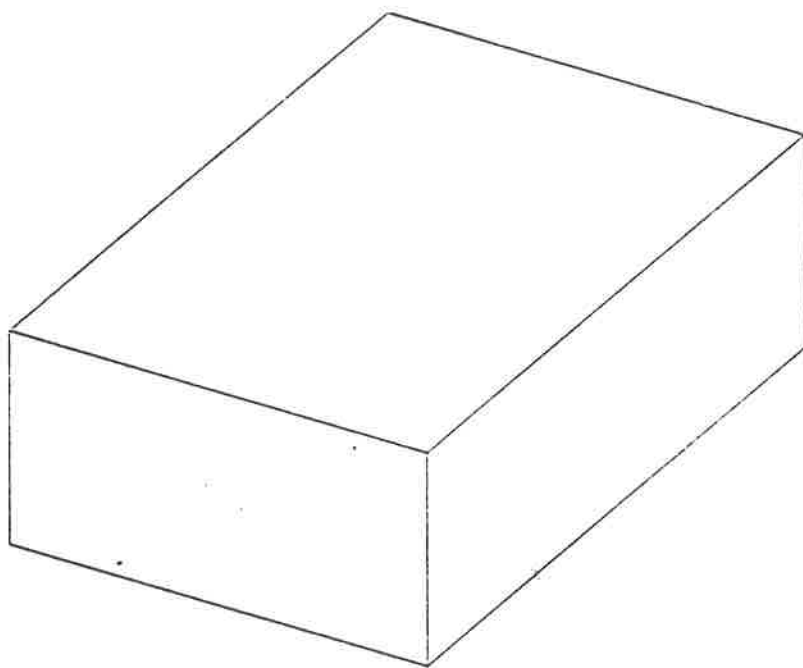


Cos B-A.

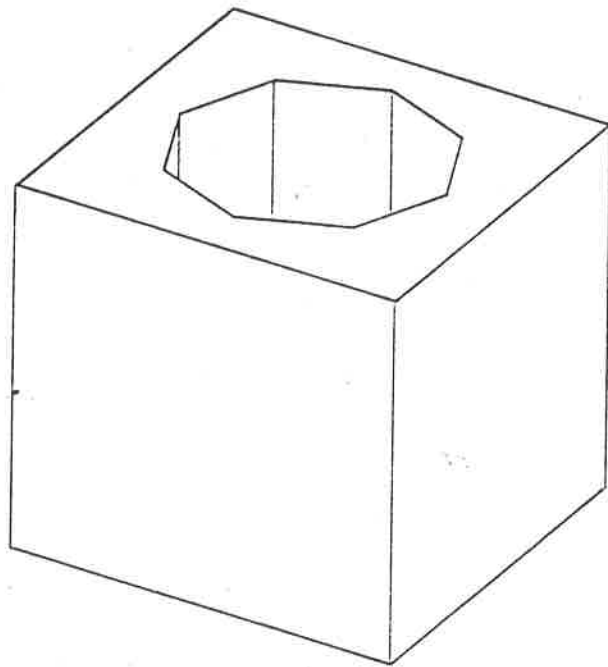
EXEMPLE 2

=====

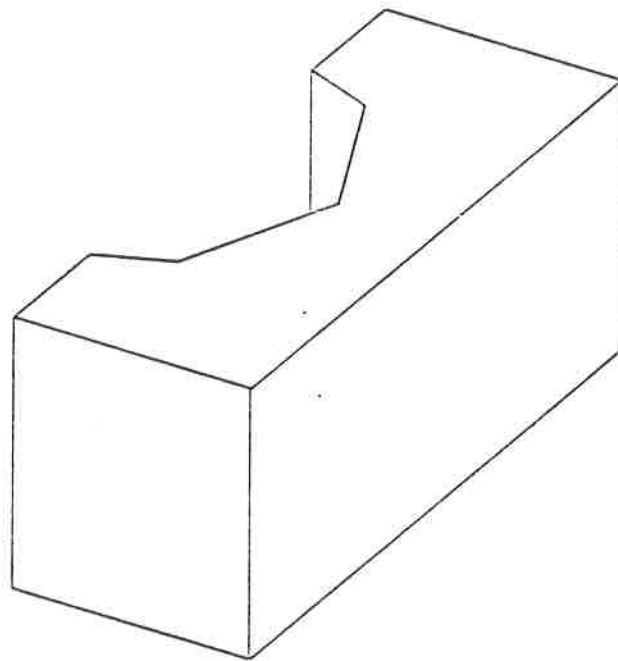




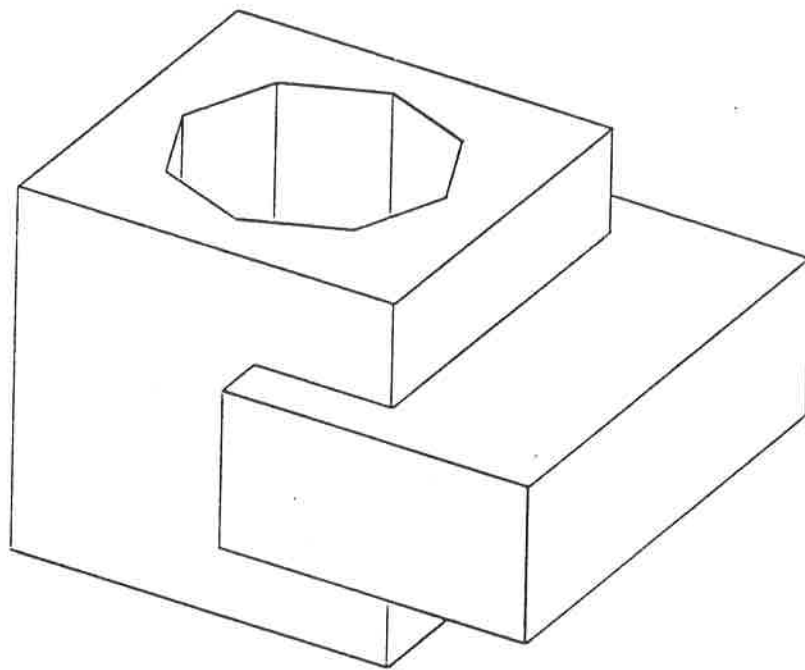
Cos B.



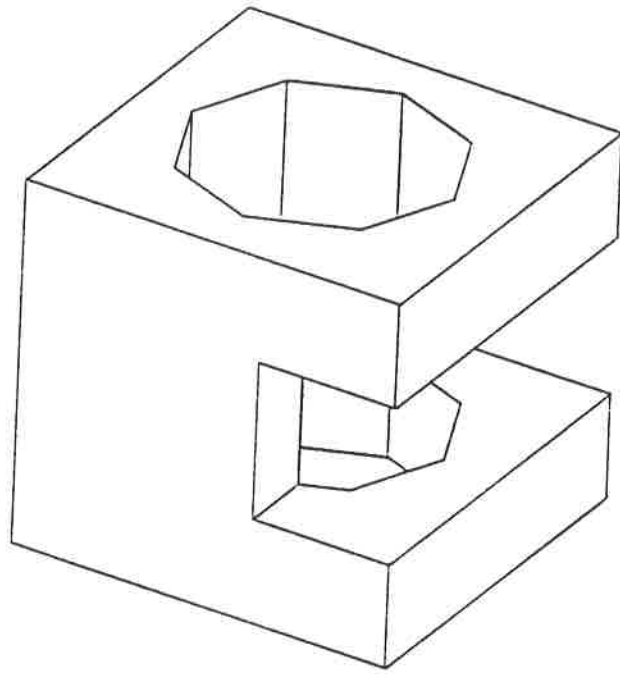
Cos A.



Cos A  $\cap$  B.



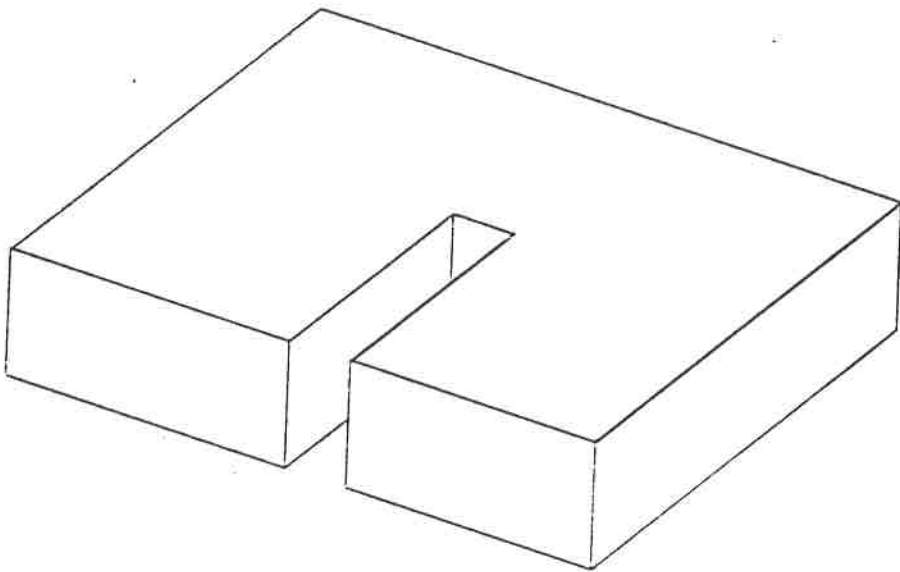
Cos A  $\cup$  B.



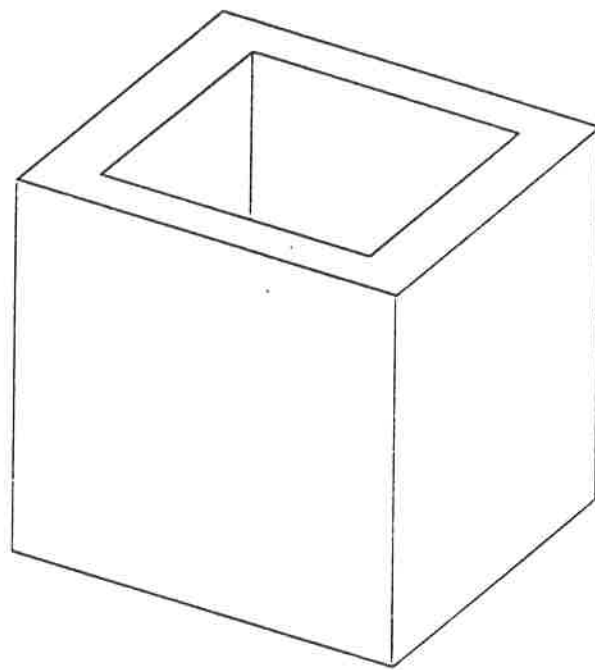
Cos A-B

EXAMPLE 3

=====

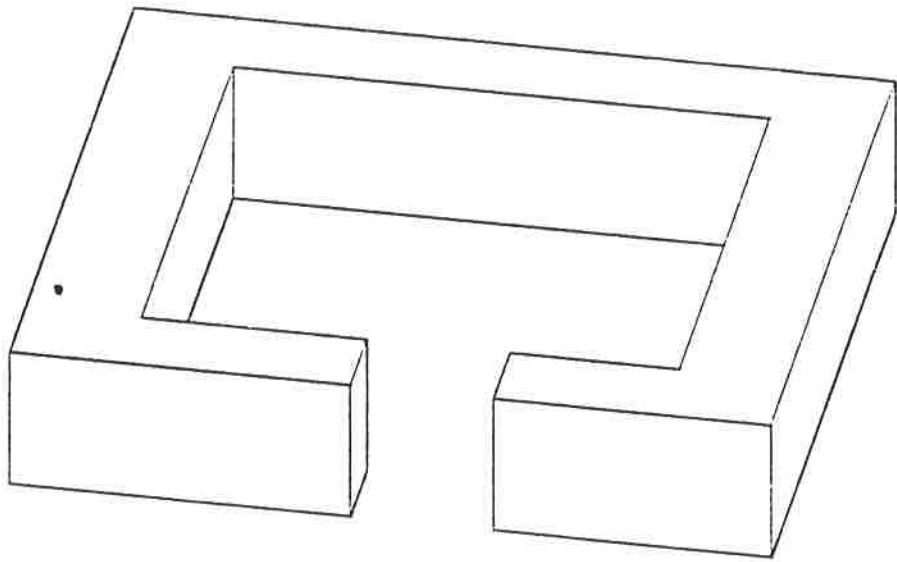


Cos B.

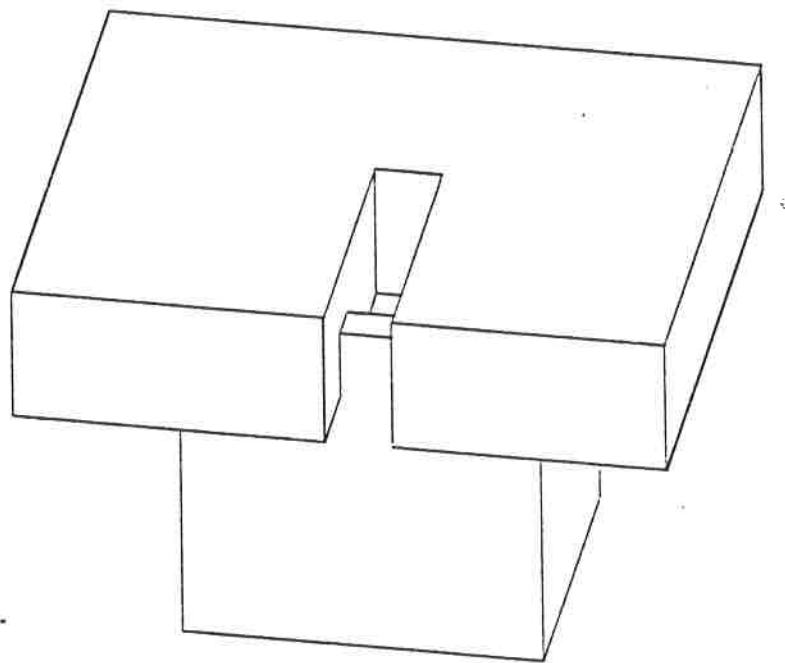


Cos A.

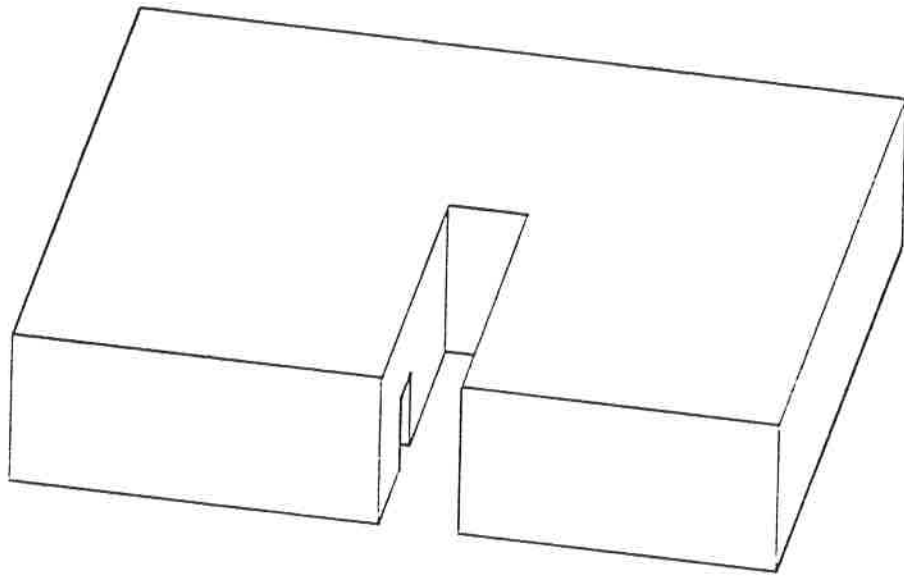




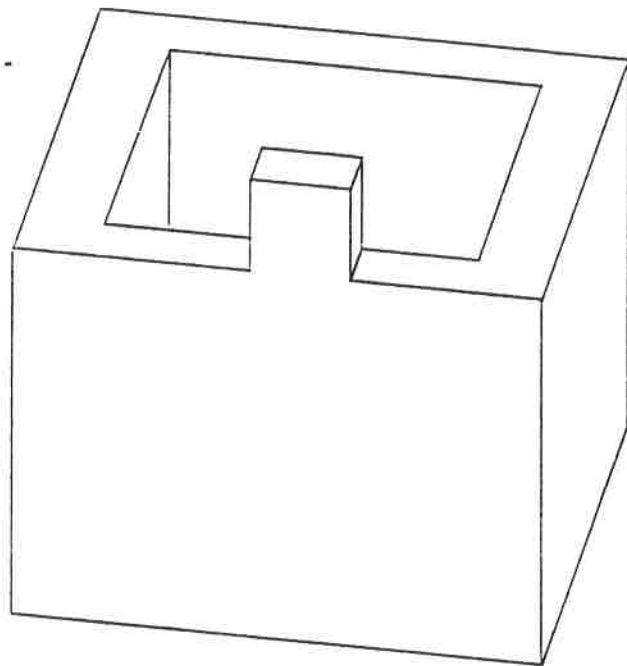
Cos  $A \cap B$ .



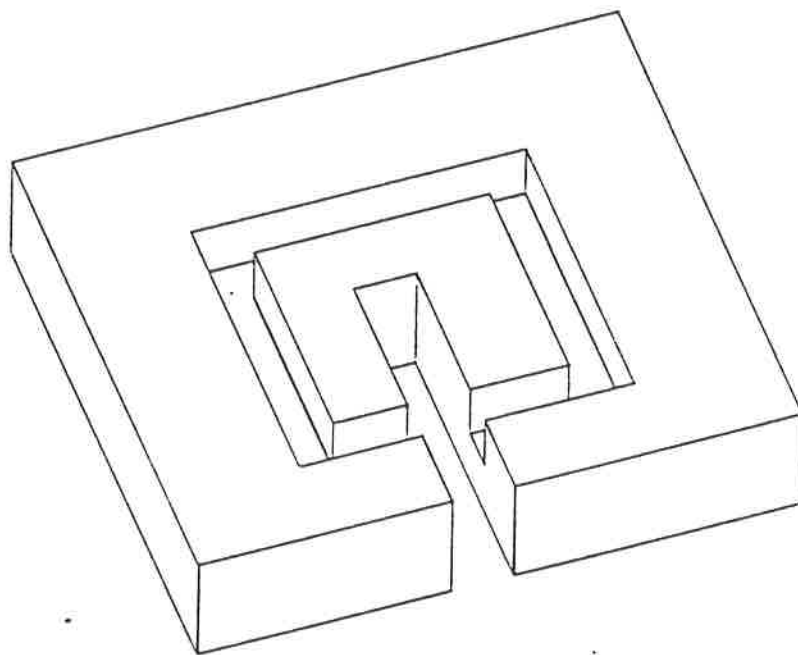
Cos  $A \cup B$ .



Cos B-A.



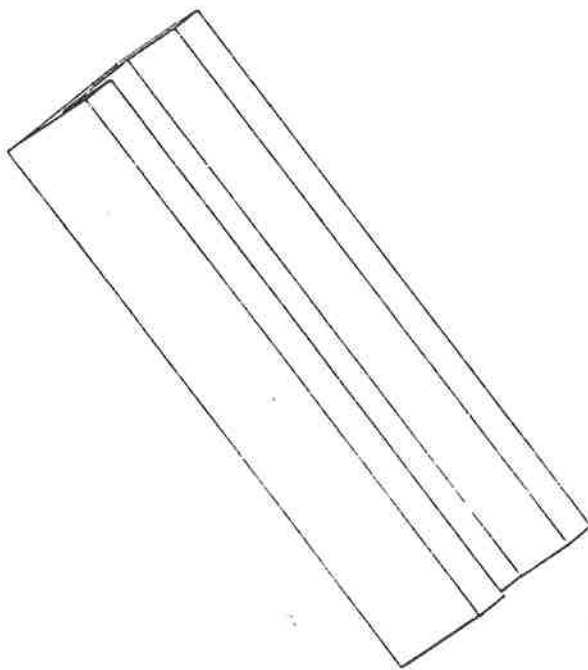
Cos A-B



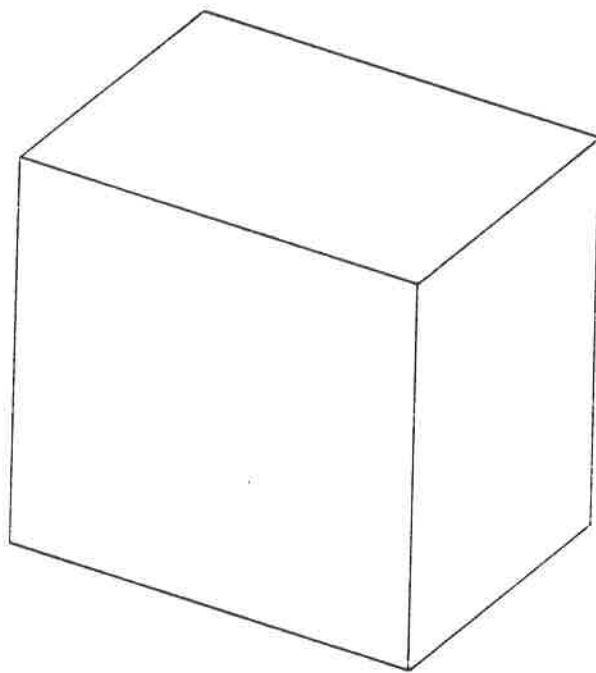
Cos B-A (Vist per sota).

EXEMPLE 4

=====

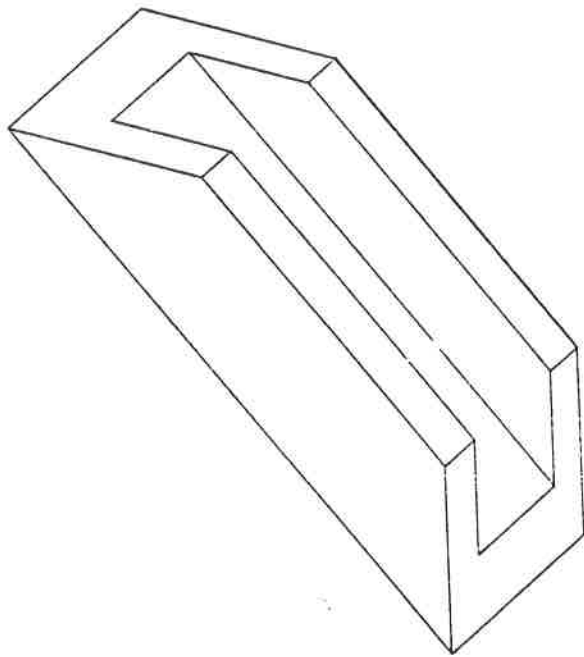


Cos B.

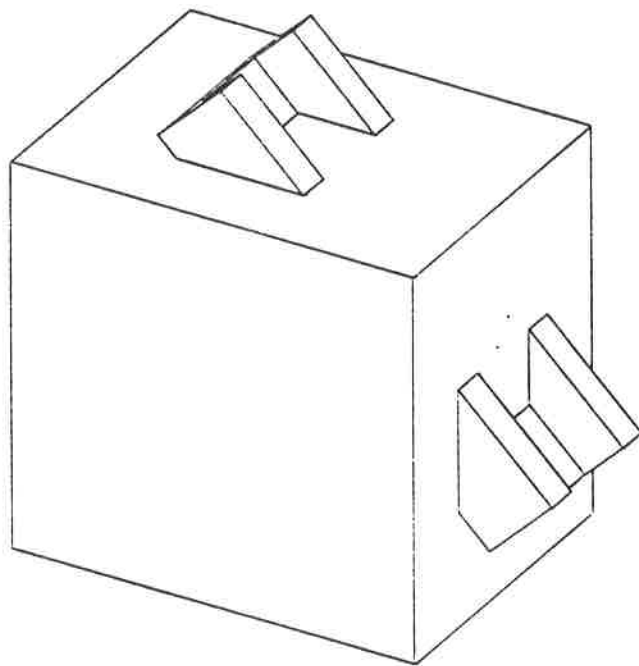


Cos A.

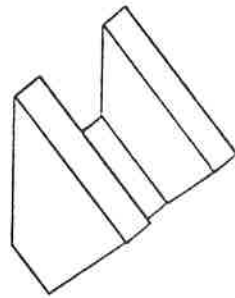
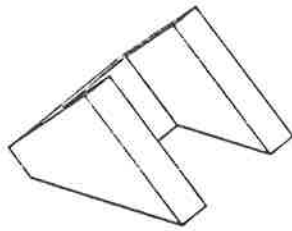




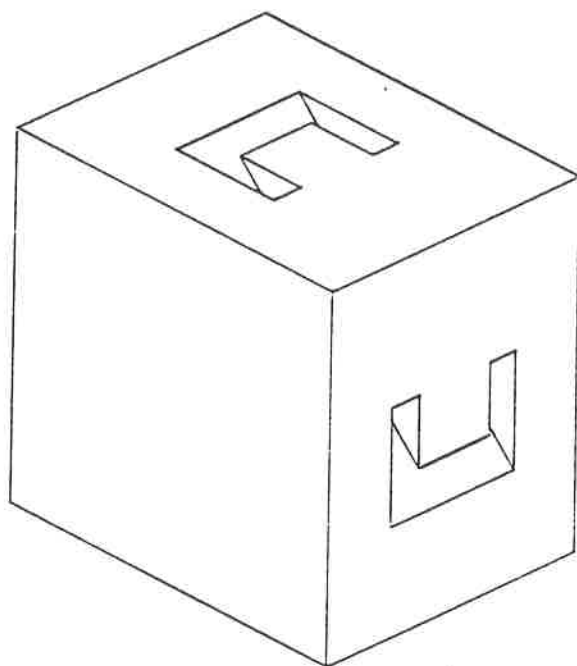
Cos A  $\cap$  B.



Cos  $A \cup B$ .



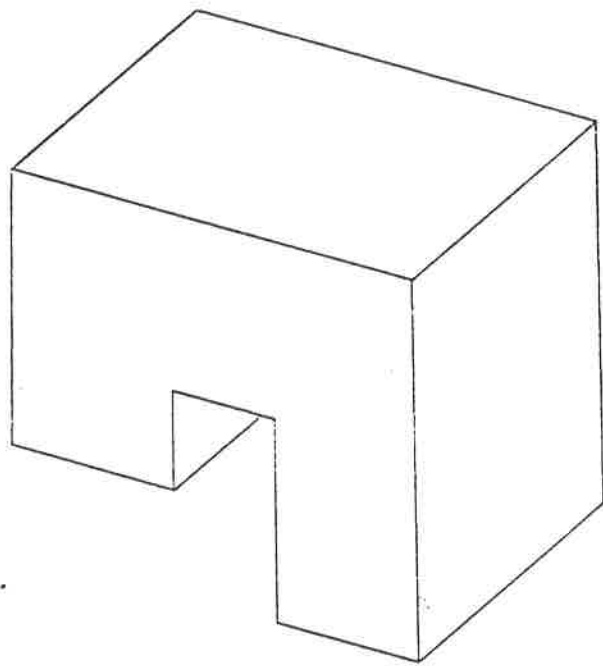
Cos B-A.



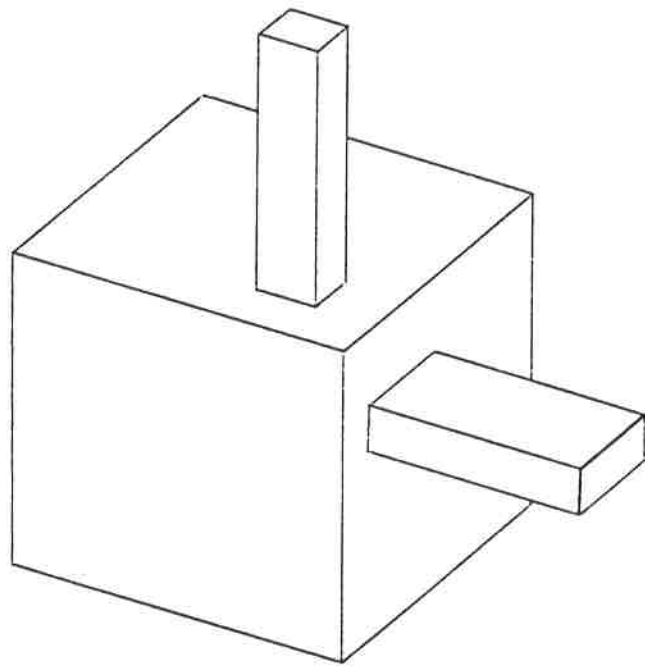
Cps A-B

EXEMPLE 5

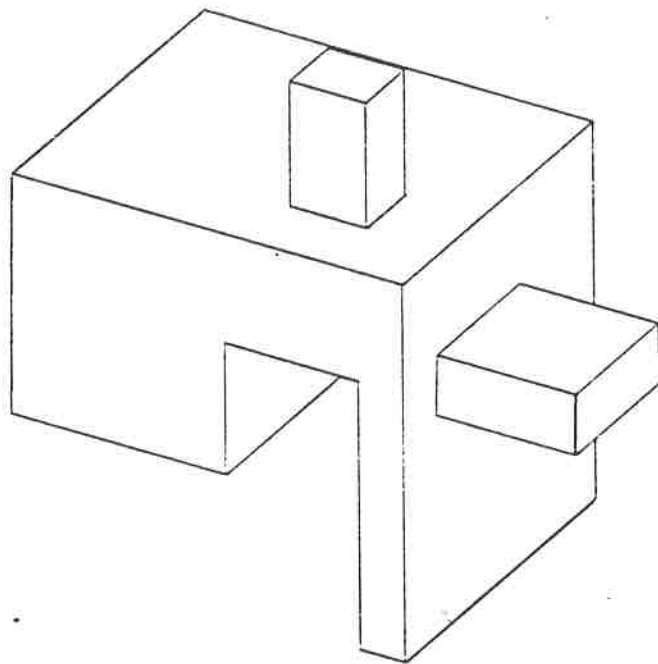
=====



Cos B

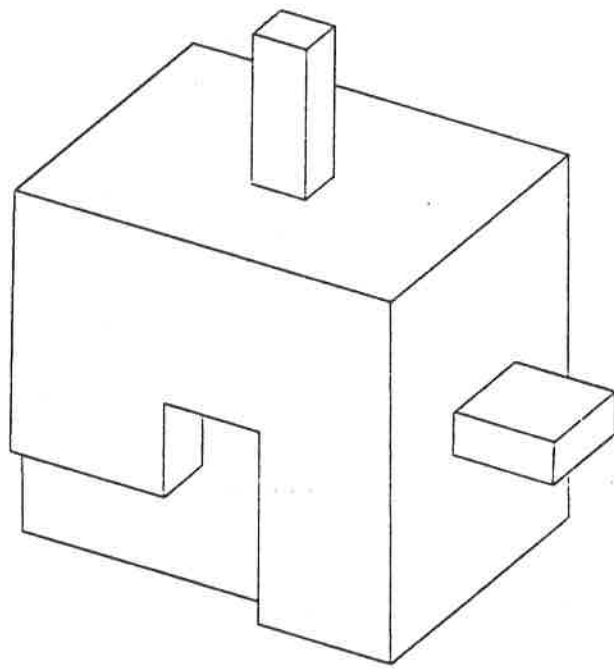


Cos A.

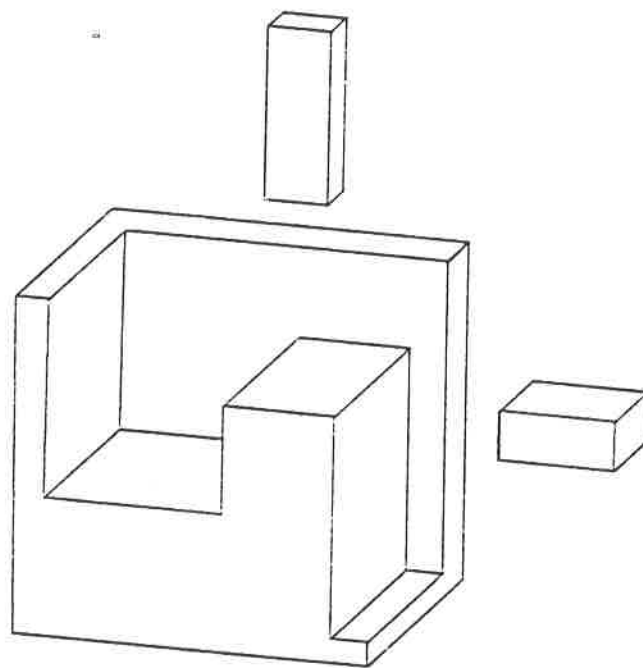


Cos A  $\cap$  B.

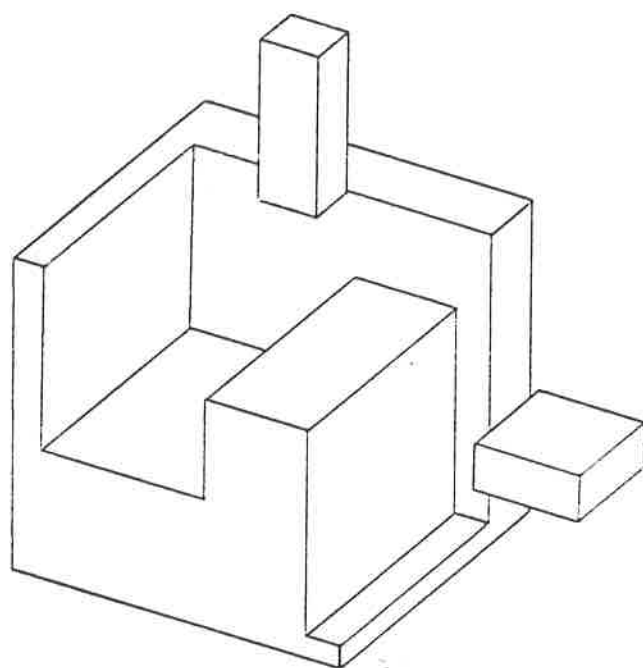




Cos A  $\cup$  B.



Cos A-B (girant la vista anterior cap a la dreta i cap amunt).



Cos A-B.

REFERENCIES

- 1 REQUICHA, A.G. "Representations for Rigid Solids: Theory, Methods, And Systems". Computing Surveys, Vol. 12, n°4, Dec. 1980.
- 2 AYALA, D.; BRUNET, P.; JUAN, R.; NAVAZO, I. "Object Representation by Means of Non-Minimal Division Quadrees and Octrees" A.C.M. T.O.G. Gener-85.
- 3 BRUNET, P; NAVAZO, I. "Geometric Modelling Using exact octree Representation of Polyhedral Objects" Eurographics'85.
- 4 REQUICHA, A.G.; VOELKER, H.B. "Boolean Operations in Solid Modelling: Boundary Evaluation and Merging Algorithms" Proceedings of the IEEE, Vol. 73, n°1. Gener 1985
- 5 MANTYLA, M. "Set Operations of GWB" Rep.-HTKKTKO B48, Lab. of Information Processing, Helsinki Univ. Technology, Finland 1982.
- 6 HEGRON, G. "Algorithmes Elementaires de Decoupage et Traitements de Nature Geometrique pour la Production d'Images par Ordinateur" Institut de Mathematiques et d'Informatique. Universite de Nantes. Gener 1983.
- 7 BRUNET, P.; LLISTERRI, A.; PASCUAL, J.; PUEYO, X. "Una base de dades gràfica per el disseny interactiu en 3-D sobre mini o microordinadors" CIL-83
- 8 BRUNET, P. et al. "Diseño y fabricacion por computador. Sistemas CAD, CAM, CAE." Ed. Marcombo S.A. 1986.

## Relació de Taules, Esquemes i Figures.

TAULES

pag.

Taula 1. Operacions Booleanes

ESQUEMES

Esquema 0. Estructura de dades del model de fronteres	3
Esquema 1. Estructura de dades de la sortida del procés 1 Bloc 1.	9
Esquema 2. Estructura de dades de la sortida del procés 1 Bloc 2.	9
Esquema 3. Estructura de dades de la sortida del procés 2	16
Esquema 4. " " " " " "	3 21

FIGURES

Figura 0. Interés de les operacions booleanes.	4
" 1. 2 cubs interferint.	6
" 2. Una diferència entre els 2 cubs. A-B	7
" 3. Obtenció de les arestes de tall	11
" 4. Explosió d'una cara	13
" 5. Pas d'informació al procés 3.	18
" 6. La cara 2 no té cap punt significatiu.	20
" 7. Punt interior a un sòlid.	20
" 8. Procés 4. Polígons nous.	23
" 9. Cara del cos A.	24
" 10. Operacions Booleanes dels objectes de la figura 8	24
" 11. Cara del cos A amb forat vell	25
" 12. Cara amb forats nous disjunts.	26
" 13. Cara amb forats nous no disjunts.	27

<u>INDEX</u>	pag.
1.- Introducció	1
1.1 Modelat Geomètric	1
1.2 El model de fronteres	2
1.3 Operacions Booleanes. Sistemes existents	3
2.- Descripció general del mètode que es presenta	5
3.- Procés 1	8
3.1 Introducció	8
3.2 Algorisme	10
3.3 Comentaris	12
4.- Procés 2	12
4.1 Introducció	12
4.2 Mètode i algorisme	13
4.3 Estructura de dades de la sortida	15
5.- Procés 3	16
5.1 Introducció	16
5.2 Mètode i algorisme	16
5.3 Estructura de dades a la sortida	20
6.- Procés 4	21
6.1 Introducció	21
6.2 Mètode i algorisme	26
7.- Exemples	30
8.- Referències	65
9.- Relació de Taules Esquemes i figures.	66