

ELS MÈTODES DE CàLCUL DE VISIBILITAT I LA SEVA PARAL·LELITZACIÓ: ORIGEN, ÚS I APLICACIONS

J. Surrell (joan.surrell@ima.udg.es)

Institut d'Informàtica i Aplicacions, Universitat de Girona, Campus Montilivi, s.n., 17003 Girona.

RESUM

Aquest article conté un estudi dels mètodes de càlcul de visibilitat usats en gràfics per computador (centrat principalment en les tècniques d'escombratge) i la seva paral·lelització, tant en els algorismes de visualització tridimensional (origen d'aquests mètodes) com en altres que utilitzen el càlcul de la visibilitat com un element més del seu procés (animació i radiositat). Es presenta el càlcul de visibilitat com un element de la cadena de visualització tridimensional, es fa una exposició de les diferents arquitectures paral·leles existents així com dels problemes que comporta la paral·lelització dels algorismes, es presenta un estat de la qüestió en visualització tridimensional, animació i radiositat i s'aporta un estudi comparatiu de les diferents possibilitats exposades.

RESUMEN

Este artículo contiene un estudio de los métodos de cálculo de visibilidad usados en gráficos por computador (centrado principalmente en las técnicas de barrido) y su paralelización, tanto en los algoritmos de visualización tridimensional (origen de estos métodos) como en otros que utilizan el cálculo de la visibilidad como un elemento más de su proceso (animación y radiosidad). Se presenta el cálculo de visibilidad como un elemento de la cadena de visualización tridimensional, se hace una exposición de las diferentes arquitecturas paralelas existentes así como de los problemas que representa la paralelización de los algoritmos, se presenta un estado del arte sobre visualización tridimensional, animación y radiosidad y se aporta un estudio comparativo de las diferentes posibilidades presentadas.

ABSTRACT

This paper focuses on visibility computation methods used in computer graphics (mainly in scan-line techniques) and related parallel algorithms. The survey includes 3D visualization techniques and others algorithms based on visibility computation methods (like computer animation and radiosity). The paper has an introduction to the visibility computation methods used in rendering algorithms, a discussion on parallel architectures and their programming problems and the state of the art on parallel rendering, parallel animation and parallel radiosity, each one with a comparative analysis.

Keywords: Rendering, Scan-line, Parallel Algorithms, Computer Animation, Radiosity.

INTRODUCCIÓ

Dins de la informàtica gràfica tridimensional, els algorismes de càlcul de visibilitat ocupen un lloc destacat a causa, principalment, de la gran quantitat de recursos (espai i temps) que requereixen. Tot i que els primers algorismes daten de fa més de

25 anys [Bou 70] [SSS 74], aquest tema no ha perdut la seva importància, com ho demostra el nombre de treballs fets des d'aleshores i els que s'estan fent actualment en una gran diversitat d'àmbits. Els treballs fets en els darrers temps han anat encaminats a obtenir algorismes per tractar diferents geometries (quàdriques, superfícies esculpides) [Pue 86], a l'obtenció de realisme en la imatge (bàsicament traçat de raigs [Whi 80] i radiositat [Pue 91]), així com per millorar-ne l'eficiència [GGH 88] [Whi 92].

S'han obtingut resultats notables en diversos aspectes, encara que el principal problema que existeix és la manca de mètodes que proporcionin una velocitat de sortida prou elevada, idealment apta per fer el procés en temps real (més de 10 imatges per segon). L'ús de tècniques paral·leles pot ser una de les maneres d'aconseguir-ho, encara que, fins al moment, els resultats obtinguts són discrets, són només teòrics o bé s'han obtingut amb arquitectures molt especialitzades, cosa que dificulta la seva supervivència amb el pas del temps i l'evolució de la tecnologia.

El procés de visualització descriu els passos necessaris per convertir una escena 3D en una imatge. L'esquema considerat clàssic es troba a la figura 1. S'hi pot veure que un dels passos consisteix a fer l'eliminació de les parts amagades. Per portar-lo a la pràctica s'utilitzen els mètodes de càlcul de visibilitat.

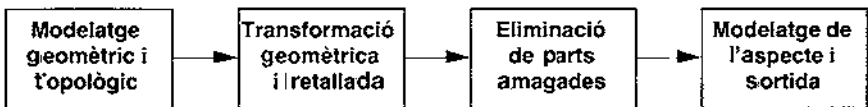


Figura 1. Esquema bàsic d'un algoritme de visualització.

Una classificació dels mètodes de càlcul de la visibilitat pren com a referència el lloc geomètric on es realitza l'ordenació geomètrica dels elements de l'escena per determinar-ne els visibles i els no visibles [Pue 86]. Així apareixen mètodes d'una dimensió (*ray-casting*, *z-buffer*...) que fan el càlcul de la visibilitat elemental comparant punts d'una mateixa línia; els mètodes de dues dimensions o d'escombratge (Watkins, pila, creuaments...) que realitzen el càlcul de visibilitat elemental comparant segments d'un mateix pla, i, finalment, els mètodes de tres dimensions (listes de prioritat...) que realitzen el càlcul de visibilitat elemental comparant elements a l'espai.

Les tècniques de càlcul de visibilitat originàries de la cadena de visualització s'utilitzen per resoldre altres problemes, alguns dels quals també estan relacionats amb la visualització. Es poden destacar les tècniques d'animació (visualitzar imatges successives per crear sensació de moviment), el traçat de raigs (per determinar l'arbre de raigs visibles en un cert punt) [Whi 80] i la radiositat (per determinar la visibilitat dels diferents subelements de l'escena o càlcul dels factors de forma) [GTG 84]. En tots aquests àmbits s'han adaptat les tècniques conegudes per resoldre el problema de càlcul de visibilitat específic de l'àmbit en qüestió.

En els apartats que segueixen es fa una exposició dels conceptes bàsics del paral·lelisme tant en el vessant d'arquitectures existents com en els problemes que

presenta el desenvolupament de programes que aprofitin les possibilitats que tenen els sistemes paral·lels. Es presenta un estat de la qüestió en visualització tridimensional per escombratge, animació i radiositat acompanyat d'un estudi comparatiu de les diferents tècniques presentades, i es finalitza l'article aportant unes conclusions de l'estudi fet.

PARAL·LELISME

Concepte i necessitat

Reben el nom de màquines paral·leles aquelles que disposen de diversos processadors que realitzen operacions a la vegada. La necessitat de disposar de diversos processadors sorgeix de la manera com funciona l'arquitectura clàssica de les màquines actuals. En aquesta arquitectura les dades processades han de passar per la unitat aritmèticològica, cosa que comporta un coll d'ampolla important en aquests sistemes i que limita en gran manera la quantitat d'informació processada. Per reduir aquesta forta limitació s'han desenvolupat sistemes amb diversos processadors que treballin alhora: són les arquitectures paral·leles.

Arquitectures paral·leles

Hi ha diverses maneres de classificar les arquitectures paral·leles. Una de les primeres classificacions que va existir [Fly 66] es basava en la multiplicitat d'instruccions i de dades que pot tractar la màquina. Així s'obtenen quatre tipus d'ordinadors:

- **SISD** (*Single Instruction, Single Data*), màquines que poden processar en un moment determinat una única instrucció sobre una única dada. Entren en aquesta categoria les màquines amb arquitectura de Von Neumann, és a dir, màquines no paral·leles (seqüencials) que corresponen a la majoria de màquines existents.
- **SIMD** (*Single Instruction, Multiple Data*), màquines que poden executar una única instrucció en cada instant, però poden aplicar-la simultàniament a moltes dades. Són màquines que disposen d'una gran quantitat de processadors, generalment molt elementals.
- **MISD** (*Multiple Instruction, Single Data*), màquines que poden processar diverses instruccions d'un únic flux de dades. És un model teòric però no tots els autors estan d'acord en la seva existència pràctica i, en cas d'acceptar-lo, quins sistemes s'hi adapten.
- **MIMD** (*Multiple Instruction, Multiple Data*), màquines que poden fer diverses instruccions sobre diversos conjunts de dades diferents. Les màquines d'aquesta categoria estan fetes com un conjunt de processadors SISD interconnectats de manera que poden intercanviar informació (MIMD amb pas de missatges) o compartir-la (MIMD de memòria compartida).

La classificació anterior és molt antiga (en termes relatius a la història de la informàtica). Malgrat això, conserva bona part de la seva validesa.

Problemes de la paral·lelització

La paral·lelització d'algoritmes comporta problemes específics a causa de les particularitats dels sistemes. Aquests problemes depenen molt de l'arquitectura elegida.

A [Luc 90] es pot trobar una recopilació sobre paral·lelisme en gràfics. Presenta un recull molt complet de diverses tècniques utilitzades per paral·lelitzar diversos aspectes de la síntesi d'imatges, com ara dibuix de primitives, càlcul de la visibilitat o realisme. És un article molt general i parla tant de *software* com de *hardware* específic. Al final hi ha un conjunt d'aspectes problemàtics de la paral·lelització d'algoritmes, aspectes que poden utilitzar-se com a criteris per avaluar els diversos algoritmes paral·lels existents.

A partir de l'estudi fet en aquest article es pot considerar que els principals problemes que cal resoldre en paral·lelitzar un algoritme són els següents:

- **Divisió de la feina**, ja sigui feta en l'àmbit del processament o en el de dades. Aquest aspecte estarà condicionat pel nombre de processadors del sistema i per la quantitat de memòria disponible.
- **Comunicació** entre processadors, que preveu aspectes de sincronització entre processos (no es poden començar certes parts sense haver-ne acabat d'altres) o problemes d'exclusió mútua (accés simultani a recursos compartits).
- **Equilibri de càrrega** entre els diferents processadors, de manera que un d'ells no realitzi molta part del procés mentre que n'hi hagi d'altres que no facin res durant la major part del temps.

Avaluació del paral·lelisme

Hi ha diverses maneres per mesurar l'eficiència d'una paral·lelització. Seguidament es recullen alguns valors que permeten mesurar el grau de paral·lelisme aconseguit en un determinat algoritme. Com sol passar quan s'estudia l'eficiència dels algoritmes, els valors que es donen estan pensats per avaluar el temps, ja que és la limitació principal que solen tenir els algoritmes.

- **Acceleració paral·lela**: pretén mesurar la millora aconseguida en una paral·lelització:

$$S = \frac{t_1}{t_N}$$

on:

S: acceleració paral·lela.

t_1 : temps que tarda l'execució de codi seqüencial.

t_N : temps que tarda l'execució paral·lela amb N processadors.

- **Rendiment paral·lel**: és l'acceleració paral·lela per processador:

$$\eta = \frac{S}{N} = \frac{t_1}{N \cdot t_N}$$

CÀLCUL DE VISIBILITAT

Algoritmes d'escombratge

Els algoritmes d'escombratge són algoritmes de càlcul de visibilitat en què la visibilitat de l'escena es determina en cadascun dels diversos plans en què és descompon

aquesta. En aquests algorismes es tracen un conjunt de plans horitzontals, un per cada línia d'escombratge de la pantalla o imatge que es vol obtenir, es determinen les interseccions dels elements de l'escena amb cadascun dels plans i, finalment, es determina quins elements són visibles en aquella línia per dibuixar-los a la pantalla.

L'esquema bàsic d'aquest tipus d'algorisme és el següent:

```

ACCIÓ Escombratge
  Pretractament dels elements de l'escena
  PER totes les línies de la imatge FER
    Càlcul del pla d'escombratge
    Càlcul dels segments visibles
    Dibuix dels segments visibles
  FPER
FACCIÓ
  
```

En aquest esquema bàsic de l'algorisme es pot observar que es realitza un pretractament dels elements que permet accelerar els càlculs posteriors. En el càlcul de les interseccions del pla d'escombratge es determinen les interseccions dels elements de l'escena amb el pla d'escombratge. Aquestes interseccions són segments a sobre del pla d'escombratge. Finalment, es fa una classificació dels diferents segments respecte a l'observador per determinar-ne la visibilitat, que pot ser total, parcial o nul·la. La llista de segments visibles resultant és la que es visualitza a la pantalla.

A partir de l'esquema de base s'han desenvolupat molts algorismes en què s'ha intentat anar millorant l'eficiència, incloent noves geometries o permetent el tractament de diferents models. Una de les millores que s'ha introduït en la major part dels algorismes és la utilització de coherència per accelerar els càlculs. S'anomena coherència el fet d'aprofitar certes propietats de l'escena o de la imatge que permeten simplificar i accelerar el càlcul de la visibilitat en algunes zones.

L'algorisme d'escombratge data del final dels anys seixanta [SSS 74]. La versió més difosa dels algorismes d'escombratge desenvolupats en aquests primers temps és deguda a Watkins. Posteriorment se n'han desenvolupat altres versions per millorar algun dels aspectes de l'algorisme (eficiència, tipus de model, tipus de geometria, entre altres). Un algorisme molt difós basat en el mètode d'escombratge línia a línia és l'anomenat de memòria de profunditat (*z-buffer*). En aquest algorisme, el càlcul de la visibilitat es realitza utilitzant una memòria de pantalla i una memòria de profunditat, que emmagatzemen les dades de la línia processada. En aquestes memòries s'hi van projectant els diferents segments dels polígons i les seves profunditats, de manera que només es té en compte la part del polígon que és al davant dels polígons representats fins al moment.

Algorismes paral·lels

L'aplicació del paral·lelisme a diferents parts de la informàtica gràfica no es pot considerar un tema nou, però sí d'actualitat, ja que cada vegada es vol aconseguir

una velocitat de representació més gran fent servir algoritmes que donin resultats més realistes però més costosos, cosa que obliga a utilitzar mètodes paral·lels per fer els diferents càlculs per obtenir la imatge. Darrerament s'han fet molts estudis sobre l'aplicació del paral·lelisme a les etapes de realisme de la cadena de visualització, amb la qual cosa es poden trobar gran quantitat d'algoritmes paral·lels per traçat de raigs o per radiositat [IAB 91] [D&S 92] [Vil 92]. Comparativament, no hi ha gaires estudis sobre la utilització d'algoritmes paral·lels en l'eliminació de parts amagades. A més, entre els pocs existents, n'hi ha molts que daten de fa alguns anys i ja són una mica obsolets a causa de l'evolució tecnològica, i n'hi ha d'altres que s'han desenvolupat teòricament, sense arribar mai a implementar-los, amb la consegüent falta de dades que permetin fer-ne una avaluació.

A [MCE 94] es presenta una proposta de classificació dels algoritmes paral·lels d'eliminació de parts amagades basada en el lloc on es realitza la reordenació paral·lela dels elements dins de la cadena de visualització.

El procés de visualització parteix d'un conjunt d'elements tridimensionals (escena), als quals s'aplica una transformació geomètrica i, a continuació, es dibuixen a la pantalla eliminant-ne les parts no visibles. Tant la transformació geomètrica com el dibuix són etapes on és factible aplicar paral·lelisme per accelerar el procés. Com que qualsevol element de l'escena pot anar a qualsevol part de la pantalla, cal establir estratègies de divisió dels elements que permetin fer les dues operacions paral·lelitzables. De fet, es pot considerar la visualització com una ordenació dels elements que es reflecteix a la pantalla en forma d'imatge. El lloc i la manera com es faci aquesta ordenació (i la divisió que comporta) permeten distingir tres tipus d'algoritmes paral·lels de visualització: d'ordenació inicial o predivisió (*sort-first*), d'ordenació intermèdia o indivisió (*sort-middle*) i d'ordenació final o postdivisió (*sort-last*). Els algoritmes presentats a continuació (i molts dels estudiats dins l'apartat d'animació) es poden estudiar segons aquesta classificació.

Un dels primers treballs de paral·lelització del càlcul de visibilitat és [K&G 79]. En aquest treball, els autors volen paral·lelitzar dos algoritmes de càlcul de visibilitat (Watkins i Warnock) utilitzant una arquitectura teòrica (figura 2), composta d'un processador encarregat de gestionar tot el procés i un cert nombre de processadors esclaus que realitzen un mateix algoritme sobre diferents parts de la pantalla (predivisió). Els processadors esclaus es poden veure com a processadors MIMD de memòria distribuïda, estan comunicats entre ells i amb el processador gestor mitjançant un bus i comparteixen entre ells la memòria de la imatge, subdividida en parts iguals per evitar problemes d'exclusió mútua. Tots els processadors executen el mateix algoritme sobre diferents zones de la imatge. Com assenyalen els autors, aquesta paral·lelització comporta una pèrdua de coherència. Tot i ser un dels primers treballs fets sobre el tema, ja tracta el problema que comporta la comunicació d'informació entre els processadors i la necessitat de fer un bon equilibri de càrrega entre els processadors per tal de tenir-los el màxim d'ocupats possible.

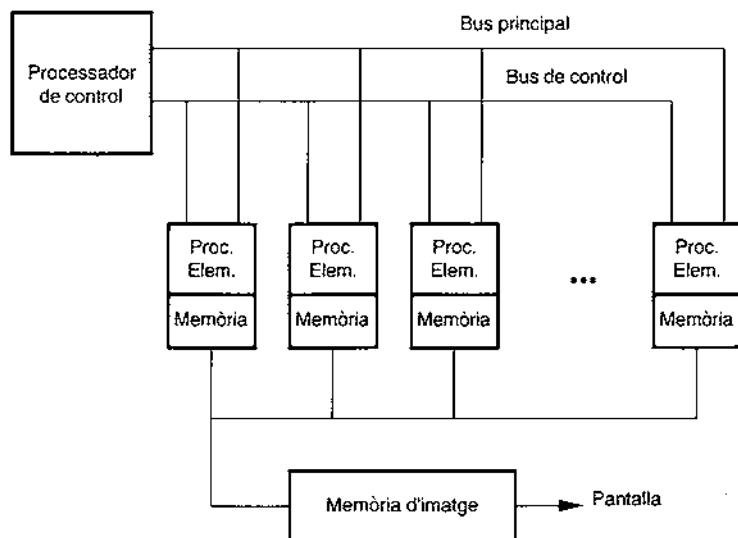


Figura 2. Sistema utilitzat a [K&G 79].

[C&R 81] van desenvolupar un algoritme per fer la paral·lelització del càlcul de visibilitat utilitzant una màquina amb diversos processadors i memòria distribuïda. L'algoritme paral·lelitzat torna a ser Watkins. La subdivisió de tasques s'aconsegueix fent una divisió de la pantalla en zones disjunctes (predivisió), cosa que permet tenir imatges locals fàcilment unibles per fer la imatge global. Presenta dues tècniques de fer la subdivisió (per bandes horitzontals i per reticulat, figura 3) que, en vista dels resultats, no tenen gaires diferències significatives.

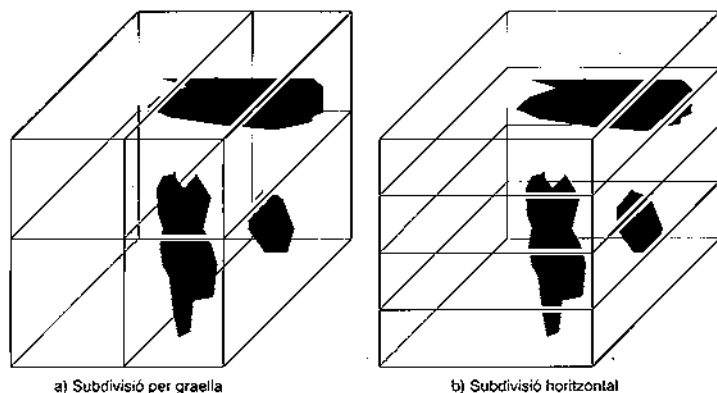


Figura 3. Subdivisions utilitzades a [C&R 81].

A [FFR 83] es presenta un mètode per paral·lelitzar l'algoritme d'escombratge utilitzant un sistema multiprocessador amb memòria compartida. Per accelerar el procés, aquest equip (anomenat ultracomputador) utilitza memòria *caché* en els processadors. Per interconnectar els processadors i la memòria hi ha una xarxa de commutadors que assegura que cada processador té accés exclusiu a una part de la memòria. Així s'eviten problemes d'exclusió mútua, però no d'accés a memòria, ja que es poden provocar bloquejos si tots els processadors volen accedir a la mateixa zona (pantalla o escena).

Pel que fa a la subdivisió de la feina, cada processador realitza el mateix algoritme (escombratge per *z-buffer*) sobre una part de la pantalla (predivisió). Per simplificar el procés, prèviament al càlcul de la visibilitat s'ha fet una subdivisió dels polígons en trapezis. En el mateix article es presenten dos mètodes per millorar la qualitat de les imatges obtingudes utilitzant tècniques paral·leles d'anti-aliasing. Tots dos mètodes es basen en la subdivisió de cada píxel per conèixer-ne més exactament el color (figura 4).

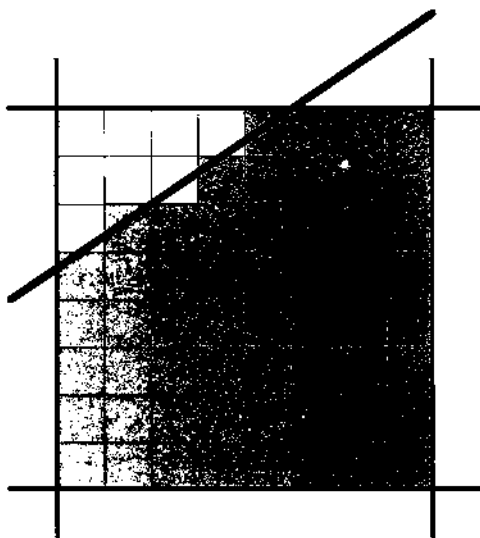


Figura 4. Mètode utilitzat a [FFR 83] per disminuir l'"aliasing".

Una de les primeres implementacions no seqüencials d'algoritmes de càlcul de la visibilitat sobre màquines de propòsit general es pot trobar a [D&W 87]. En aquest document es presenta la vectorització de l'algoritme d'escombratge per *z-buffer* utilitzant una màquina Convex de gamma baixa. En l'article es fa un estudi de l'algoritme, la seva descomposició en parts i els possibles llocs on es pot vectoritzar el procés. Per optimitzar la vectorització de l'algoritme d'escombratge es proposa la substitució d'estructures de dades basades en punters per les seves equivalents estàtiques basades en taules i l'intercanvi d'ordre de niament d'alguns bucles de manera que s'aprofitin millor les possibilitats de vectorització que té la màquina. No es pot considerar aquesta implementació com un exemple clar de cap de les tècniques de paral·lelització.

Una de les implementacions més completes d'algoritmes d'eliminació de parts amagades sobre arquitectures de propòsit general pot trobar-se a [The 87], on es presenta una arquitectura de propòsit general de tipus híbrid SIMD-MIMD anomenada Disputer. Un Disputer (figura 5) està constituït per una xarxa de Transputers (MIMD) i un DisArray (SIMD) i es pot programar fent servir un llenguatge basat en el model CSP [Hoa 78].

Fent una simplificació del funcionament del sistema es pot dir que la part MIMD fa el preprocessament dels polígons per visualitzar i de l'eliminació de les parts amagades, i la part SIMD s'encarrega de la visualització en si mateixa. El conjunt d'algoritmes presentats permet realitzar l'eliminació de parts amagades d'escenes descrites utilitzant polígons, aplicant-hi un cert grau d'anti-aliasing. Es fa un estudi força complet sobre l'eficiència del sistema, analitzant fins i tot la problemàtica que pot comportar el traspàs d'informació entre els processadors. El mètode utilitzat no s'adapta de manera exacta a cap de les possibles paral·lelitzacions, sinó que es pot considerar un híbrid entre elles.

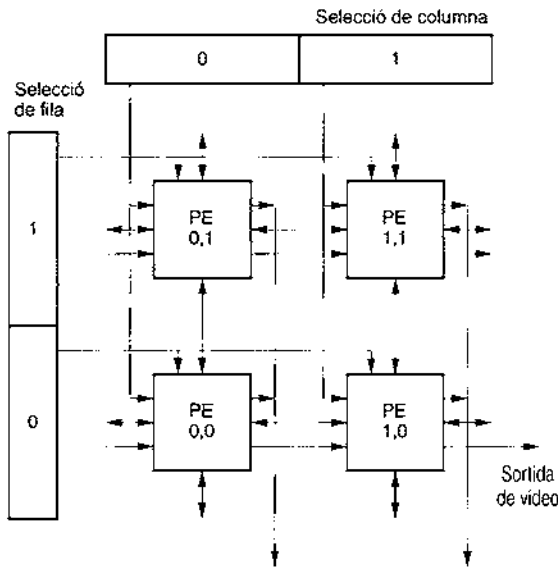


Figura 5. DisArray de 2x2 elements de procés [Teo 87].

A [Rob 88] es presenta la paral·lelització de l'algoritme d'escombratge per z-buffer. En aquesta paral·lelització s'utilitza un hipercub de fins a 16 nodes (4 dimensions). L'algoritme presentat realitza una part del càlcul (les transformacions geomètriques i els pretractaments) de manera seqüencial utilitzant el node gestor de l'hipercub. La paral·lelització correspon a l'eliminació de parts amagades en si i es fa a la pantalla (indivisió). No es realitza cap tipus d'ombreig (model d'il·luminació). Per equilibrar la càrrega dels diferents processadors es du a terme una doble distribució de les dades, la primera uniforme pel que fa a àrea de la pantalla i la segona més uniforme en nombre de cares per processador (àrees no iguals, figura

0	125	91	0
29	225	212	25
71	161	156	58
6	38	27	4

a) Zones i polígons a l'inici

29	62	91	87
	63	106	
77	113	108	87
	112	78	
77	80	78	87
	81	78	
77	38	27	87
	38	27	

b) Zones i polígons després de fer la redistribució

Figura 6. Exemple d'un balanç de càrrega [Rob 88].

Un altre article important és [H&B 92], on s'implementen diversos algoritmes de càlcul de la visibilitat en paral·lel (subdivisió recursiva, escombratge, *z-buffer* i l'algoritme del pintor). L'arquitectura triada és una cadena de Transputers (arquitectura MIMD de memòria local, figura 7) en què cada element de la cadena realitza el processament d'una part de la pantalla, si bé en alguns dels algoritmes (*z-buffer* i pintor) s'ha desenvolupat una estratègia de paral·lelització sobre els polígons (post-divisió).

El resultats obtinguts no són gaire bons en les paral·lelitzacions sobre els polígons a causa del coll d'ampolla que provoca l'accés en exclusió mútua a la pantalla. Entre les estratègies de paral·lelització sobre la pantalla, el *z-buffer* i l'algoritme d'escombratge són les que presenten una millor linealitat en canviar el nombre de processadors (d'1 fins a 128 processadors), amb valors més propers a la unitat com més gran és l'escena utilitzada per avaluar l'algoritme.

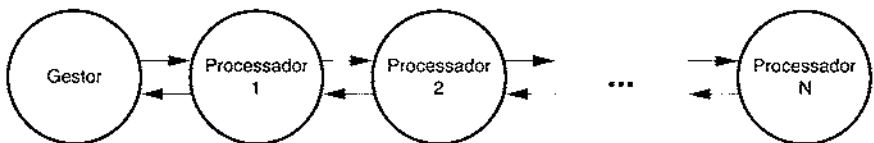


Figura 7. Topologia d'interconnexió dels Transputers [H&B 92].

D'aquest article es pot destacar el fet que fa un estudi del rendiment paral·lel dels algoritmes presentats i avalua a la pràctica diversos paràmetres: el nombre de polígons de l'escena, el nombre de processadors, l'ús de coherència en algoritmes paral·lels, entre altres.

Comparació d'algoritmes

En comparar els algoritmes presentats anteriorment, es poden destacar els aspectes següents:

- La majoria de les propostes utilitzen sistemes MIMD de memòria distribuïda, amb l'excepció de [FFR 83], que fa servir MIMD de memòria compartida, [D&W 87], que usa una arquitectura vectorial, i [The 87], que utilitza una arquitectura híbrida SIMD/MIMD.
- Hi ha diversos articles que corresponen a tècniques de predivisió [K&G 79] [C&R 81] [FFR 83], un de tècniques d'indivisió [Rob 88] i un de postdivisió [H&B 92] (dues de les referències no es poden agrupar sota cap de les tres tècniques).
- La majoria dels mètodes no preveuen cap tècnica d'equilibri de càrrega entre els processadors. L'única excepció és [Rob 88].
- Com a algorisme de base s'utilitza Watkins [K&G 79] [C&R 81] o *z-buffer* [FFR 83] [D&W 87] [The 87] [Rob 88] [H&B 92].
- La majoria dels algoritmes destaquen la pèrdua d'ús de coherència que comporta la paral·lelització.
- Només en el darrer treball, [H&B 92], es realitza un estudi del paral·lelisme obtingut, és a dir, acceleració paral·lela, equilibri de càrrega, rendiment paral·lel o altres de semblants.

ANIMACIÓ

Concepte

S'entén per animació l'obtenció d'una successió d'imatges amb una velocitat tal que permeti generar pel·lícules en temps real. Es pot considerar acceptable l'obtenció de més de 10 imatges per segon, encara que en calen entre 25 i 50 per obtenir resultats de qualitat. En cas que no es pugui aconseguir una velocitat suficient d'obtenció d'imatges, cal fer primer la generació, guardar-les i, després, visualitzar-les a la velocitat correcta. Si per fer el càlcul d'una única imatge ja cal molta quantitat de processament, per obtenir-les amb una velocitat adequada per poder fer una animació amb 25 imatges per segon, la quantitat de processament s'incrementa de manera considerable i, per tant, la necessitat d'utilitzar qualsevol tècnica per accelerar els càlculs esdevé prioritària.

Les tècniques d'animació no afegeixen més dificultat que la pròpia derivada del gran volum de sortida d'informació que han de generar. Alguns autors intenten simplificar l'obtenció d'imatges utilitzant coherència entre imatges successives, de manera que en una imatge nova només es calcula la part que ha canviat respecte de l'anterior. Això en alguns casos pot arribar a generar estalvis importants, encara que té el problema que no és una tècnica gaire aprofitable des del punt de vista de paral·lelització dels mètodes.

Algoritmes paral·lels

En aquest apartat es presenten un conjunt de referències sobre tècniques d'animació utilitzant algoritmes paral·lels. Cal tenir en compte que en molts casos es tracta de

mètodes que acceleren l'obtenció d'una imatge i que, utilitzats adequadament, poden generar una seqüència d'imatges amb una velocitat acceptable per realitzar animació en temps real. Tanmateix, no es tracta d'algoritmes d'animació en si, sinó d'algoritmes ràpids d'eliminació de parts amagades. Només s'han inclòs en la recopilació pel fet que en la presentació de la tècnica es donen resultats del seu ús en animacions.

La recopilació s'ha limitat en el temps, ja que les tècniques de paral·lelització depenen molt de l'arquitectura i aquesta canvia molt ràpidament, circumstància que dificulta molt la supervivència de mètodes específics de certes arquitectures.

La proposta feta a [SPG 93] dóna una solució paral·lela al problema del càlcul de visibilitat que es pot aplicar a sistemes MIMD o a xarxes d'estacions de treball. La tècnica proposada dóna com a sortida elements en l'espai objecte. La proposta rep el nom de barreja paral·lela en profunditat (*parallel depth-merge* o PDM). No es tracta de la paral·lelització d'un algoritme de parts amagades seqüencial, sinó que és una proposta pensada des dels seus orígens com a algoritme paral·lel que permet obtenir diferents algoritmes de càlcul de visibilitat en l'espai objecte sobre diversos sistemes MIMD. L'algoritme és ben simple: cal dividir el conjunt d'entrada en parts, aplicar-hi una eliminació clàssica de les parts amagades i fer-hi una fusió de les dades resultants. El càlcul de visibilitat de les diferents parts es realitza en paral·lel (figura 8).

Per determinar el conjunt local d'objectes visibles pot utilitzar-se qualsevol mètode d'eliminació de les parts amagades en l'espai objecte. Cal obtenir una informació addicional, anomenada silueta, per facilitar la fusió posterior. Per fer l'equilibri de càrrega es proposa fer una distribució aleatòria de les dades en subconjunts de la mateixa mida.

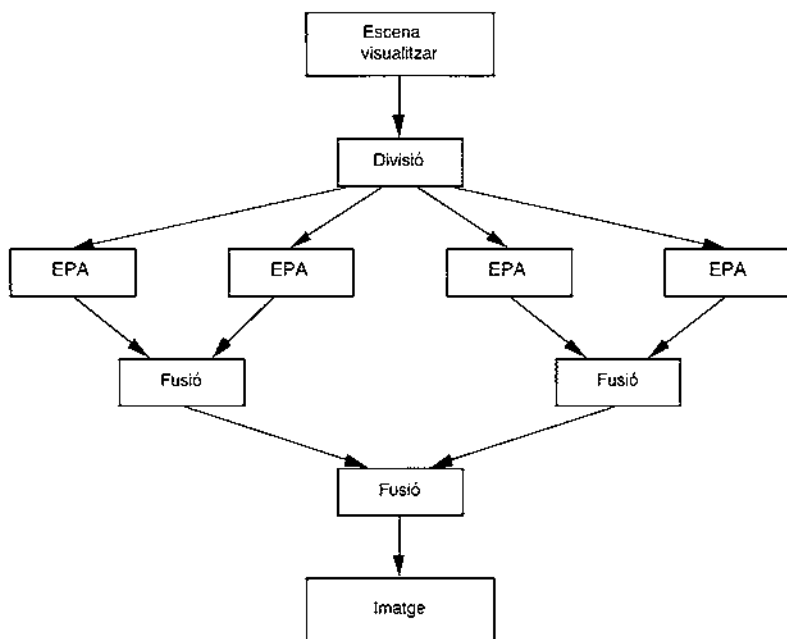


Figura 8. Divisió d'una escena entre 4 processadors usant PDM.

A [Eil 93] i [Eil 94] es presenta un algoritme paral·lel per màquines de memòria distribuïda, que permet visualitzar polígons interactivament aprofitant la coherència entre imatges successives per fer l'equilibri de càrrega entre processadors. Inicialment, les dades es reparteixen entre els processadors de manera equilibrada. Els desequilibris apareixen a la part de l'eliminació de les parts amagades. La subdivisió de la pantalla proposada utilitza una malla regular. L'equilibri de càrrega es realitza entre imatges successives, de manera que la subdivisió d'una imatge es fa a partir dels resultats de la subdivisió de la imatge prèvia.

La implementació pràctica de l'algoritme fa servir un element més per accelerar el procés: es crea una jerarquia de dos nivells en els processadors de manera que els missatges de segon nivell s'agrupen en el primer nivell i d'aquí s'envien al seu destí (figura 9):

$B_{i, j \in \{1..4\}}$ estan units entre ells

$B_{i, j \in \{1..4\}}$ estan units amb A_i

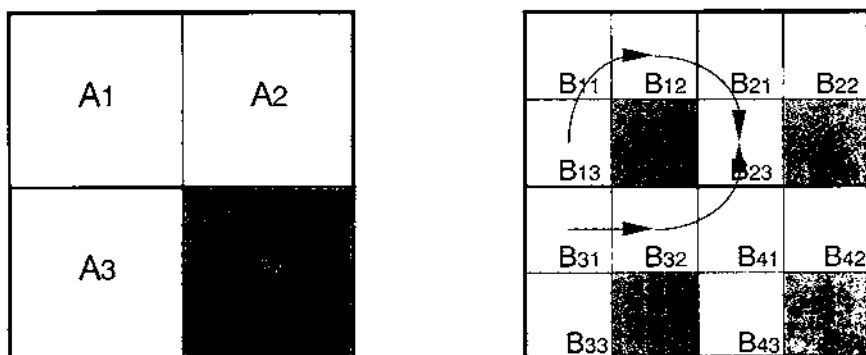


Figura 9. Jerarquia de comunicació de dos nivells [Eil 94].

L'algoritme presentat a [Whi 93] i [Whi 94] permet fer una visualització de polígons per escombratge utilitzant una divisió de la pantalla en àrees rectangulars. Aquesta divisió és la que dona àrees de perímetre mínim i preserva la coherència.

En el càlcul de la visibilitat es fa servir un conjunt de tasques que resolen els rectangles utilitzant un mètode d'escombratge per *z-buffer* amb correcció de l'*aliasing* utilitzant mostreig estocàstic múltiple. Per millorar l'equilibri de càrrega entre els processadors s'utilitza una tècnica dinàmica que consisteix a partir la feina pendent (rectangle) d'un processador ocupat amb un que ja no ho estigui (figura 10).

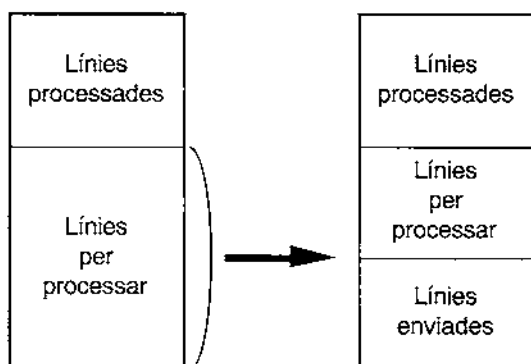


Figura 10. Subdivisió per fer l'equilibri dinàmic [Whi 94].

En el document [C&O 93] es presenta un algorisme pensat per ser aplicat a sistemes MIMD de memòria distribuïda programables per pas de missatges fent una divisió en trapezis (figura 11). S'utilitza tant el paral·lelisme en objectes com el paral·lelisme en imatges. Es fa una multiplexació de les fases inicial (transformació i retallada) i final (eliminació de les parts amagades) sobre els mateixos processadors, de manera que hi ha un millor aprofitament de la memòria, una simultaneïtat de les comunicacions amb el processament i un estalvi en la quantitat de comunicació necessària. Un esquema de l'algorisme seria:

```

MENTRE no s'acabi la feina FER
  CAS hi ha triangles locals FER
    transformar triangle
  CAS s'han rebut trapezis FER
    visualitzar trapezis
FCAS
FMENTRE

```

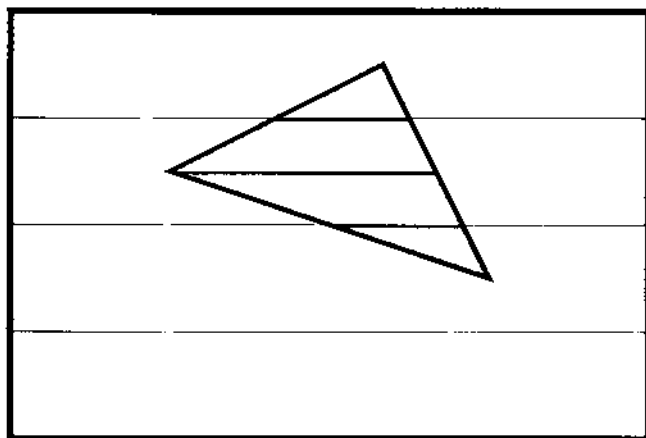


Figura 11. Divisió de la pantalla i dels polígons a [C&O 93].

A [LRN 95] es presenta un mètode per paral·lelitzar la visualització de polígons utilitzant una arquitectura MIMD amb pas de missatges. La implementació descrita es basa en un algoritme de postdivisió.

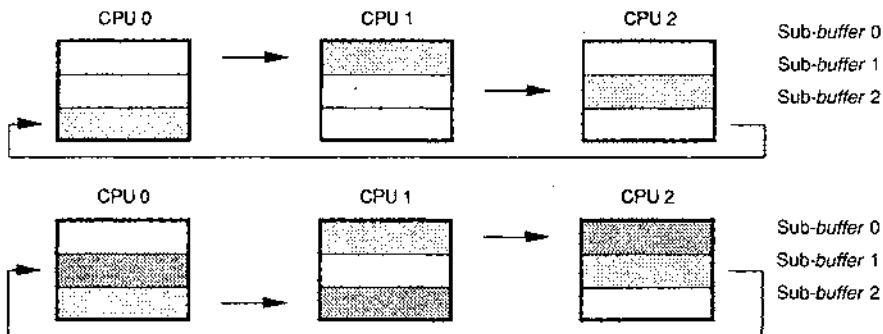


Figura 12. Fusió d'imatges a [LRN 95] (3 processadors).

Inicialment es distribueixen els polígons entre els processadors de manera dispersa. Cada processador disposarà d'una memòria en què dibuixa el conjunt de polígons que té assignats aplicant un algoritme de z-buffer i a partir de les imatges parcials es fa la composició de la imatge completa. La versió bàsica de la composició divideix els buffers de les imatges parcials en un conjunt de sub-buffers i es fusionen en paral·lel (figura 12). Per equilibrar la càrrega es proposen tècniques de tipus estàtic (prèvies al càlcul) basades en una distribució entrelaçada de l'escena, en horitzontal i en vertical.

A [Mue 95] s'estudia l'ús dels algoritmes de predivisió per obtenir sistemes gràfics de prestacions elevades, aprofitant la coherència entre imatges. Cada processador rep una part de les primitives per visualitzar i s'encarrega de visualitzar-la a la pantalla.

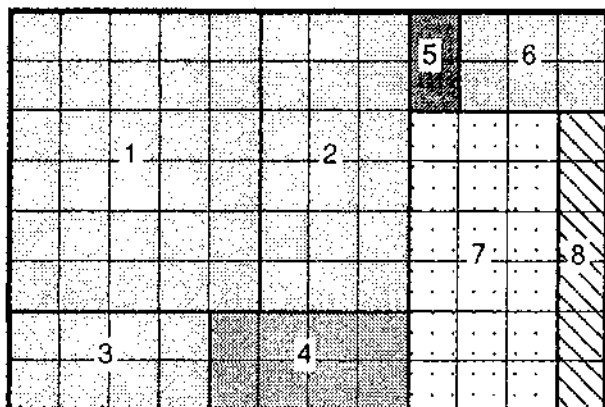


Figura 13. Subdivisió per fer l'equilibri de càrrega a [Mue 95].

Pel que fa a l'equilibri de càrrega, es proposa fer una subdivisió més gran del que cal, fer un recompte de les primitives i generar una estructura jeràrquica agafant com a fronteres els límits de les cel·les de la subdivisió (figura 13).

Comparació d'algoritmes

Els aspectes més destacables que es poden esmentar en fer la comparació dels algoritmes són:

- La majoria dels mètodes treballen amb sistemes MIMD de memòria distribuïda (de fet, només un [SPG 93] dona com a possibilitat treballar amb sistemes de memòria compartida).
- Hi ha referències que treballen amb tècniques de predivisió [Eli 93] [Eli 94] [Mue 95], amb tècniques d'indivisió [C&O 93] [Whi 93] [Whi 94] i amb tècniques de postdivisió [SPG 93] [LRN 95].
- Totes les referències preveuen algun tipus d'equilibri de càrrega estàtic i algunes proposen estratègies dinàmiques [Whi 94] [LRN 95].
- Totes les referències utilitzen com a algoritmes de base el *z-buffer*, encara que algunes [SPG 93] [Mue 95] permeten altres algoritmes de base.
- Tots els treballs fan algun tipus d'estudi quantitatiu del paral·lelisme que s'obté amb els mètodes que proposen, ja sigui en forma d'acceleració paral·lela, velocitat de sortida o, simplement, en un estudi de les comunicacions.
- Les acceleracions presentades són poc comparables entre elles, ja que s'obtenen empíricament amb escenes diferents en cada cas.

RADIOSITAT

Concepte

Les tècniques de visualització d'escenes tridimensionals han intentat aconseguir el màxim grau de realisme en l'obtenció d'imatges sintètiques. Un dels aspectes que més hi ha contribuït és l'ús de tècniques d'il·luminació global, com són el traçat de raigs [Whi 80] i, sobretot, la radiositat [GTG 84].

La radiositat, en el seu plantejament bàsic, divideix les superfícies dels objectes que componen les escenes en diferents parts (pedaços), avalua les interaccions que tenen els pedaços entre ells (càlcul dels factors de forma) i, a partir d'aquestes interaccions, determina la lluminositat de cada element. Amb els valors obtinguts és possible fer una visualització de l'escena amb un elevat grau de realisme.

La part més costosa de l'algoritme de la radiositat és el càlcul dels factors de forma, ja que cal avaluar l'efecte de cada peça sobre la resta. Els mètodes que s'utilitzaven inicialment eren de tipus matemàtic i provenien de la termotècnia. Un primer pas per millorar aquests càlculs va consistir a utilitzar tècniques de càlcul de visibilitat per avaluar els factors de forma. Es poden destacar el mètode de l'hemicub, la tècnica de traçat de raigs, els mètodes de Montecarlo, entre d'altres.

Encara que les diferents tècniques proposades per calcular els factors de forma donaven resultats cada cop millors en termes de temps de càlcul, una forta limitació de l'algoritme de la radiositat era la necessitat de guardar tots els factors de forma, aspecte que requeria l'ús d'una matriu amb cost d'espai quadràtic amb el nombre de

pedaços, $O(n^2)$. Aquesta limitació feia que qualsevol escena amb pocs milers de pedaços fos impossible de calcular, perquè necessitava guardar diversos milions de factors de forma. Per solucionar aquest problema es va desenvolupar el mètode de la radiositat progressiva [CCW 88], en què no es guardava la matriu de factors de forma, sinó que es calculava la visibilitat des del pedaç més energètic (font de llum) i, amb els valors obtinguts, es repartia l'energia que tenia disponible aquest pedaç entre la resta. Repetint el procés un nombre suficient de vegades, es pot determinar la il·luminació de cadascun dels pedaços. Un altre avantatge d'aquest mètode és l'obtenció d'una solució progressiva, cada cop més propera a la final, que es pot anar visualitzant a mesura que el procés avança. A més, com que al final s'envia una energia cada cop més petita, pot eliminar-se la part final del càlcul, a causa de la seva poca contribució al resultat definitiu.

La radiositat clàssica i la radiositat progressiva tenen el problema del gran nombre de càlculs necessaris, ja que cal calcular la interacció de cada parella de pedaços (creixement quadràtic amb el nombre de pedaços). Per millorar aquest aspecte es va desenvolupar la radiositat jeràrquica [HSA 91], en què es té en compte la posició relativa de les superfícies a l'hora de mirar les interaccions, de manera que si es tracta de superfícies que tenen molta interacció, la discretització es fa amb un grau més elevat de detall, mentre que si les superfícies es troben més allunyades, la discretització es realitza amb menys detall. D'aquesta manera es fa una discretització jeràrquica a cada superfície i cada parella de superfícies interaccionen amb un nivell més o menys elevat de la jerarquia segons la distància a què es troben.

La radiositat és un mètode que ha donat molt bons resultats amb escenes que tinguin superfícies no reflectants, mentre que els seus resultats amb escenes que tinguin superfícies reflectants són discrets. La combinació de radiositat i traçat de raigs ha permès millorar els resultats en aquests tipus d'escenes. Aquesta mena d'algoritmes tenen el nom de radiositat estesa [S&P 89]. Un altre aspecte que la radiositat no preveu és la dispersió de l'energia quan passa a través del medi on interaccionen els elements. Així, la possibilitat de visualitzar escenes amb boira o fum no dona bons resultats. Per tenir en compte aquests tipus d'escenes cal utilitzar tècniques especials, que no només avaluen la interacció entre les superfícies, sinó que també tenen en compte la interacció amb el medi, que s'ha discretitzat amb *voxels*. Aquestes tècniques s'anomenen de medis participatius.

Algoritmes paral·lels

En aquest apartat hi ha una recopilació d'articles que presenten algoritmes de radiositat paral·lela. Encara que l'estudi de la paral·lelització es vol centrar en algoritmes que fan el càlcul de la visibilitat utilitzant tècniques de *z-buffer*, també s'hi han inclòs altres tipus de tècniques, ja que moltes vegades la paral·lelització és independent del tipus de mètode sobre el qual s'apliqui.

La recopilació s'ha limitat en el temps, tal com s'ha fet a la recopilació de tècniques d'animació, ja que les tècniques de paral·lelització depenen molt de l'arquitectura i aquesta canvia molt ràpidament, circumstància que dificulta molt la supervivència de mètodes específics de certes arquitectures.

El treball presentat a [APV 91] dona les idees de base per desenvolupar un mètode alternatiu de càlcul de la visibilitat dividint l'escena en subescenes amb

l'objectiu de fer més eficient l'algoritme (esquema de divideix i venç). La divisió es realitza utilitzant el mètode de les parets virtuals (figura 14), que dóna lloc a entorns locals que es poden resoldre de manera força independent i, per tant, es poden fer servir tècniques paral·leles. L'algoritme resultant podria ser:

```

MENTRE NO Convergeix FER
  ResoldreLocalment(Entorn1+ParetVirtual)
  IntercanviarEnergia(Entorn1, Entorn2)
  ResoldreLocalment(Entorn2+ParetVirtual)
  IntercanviarEnergia(Entorn2, Entorn1)
FMENTRE
  
```

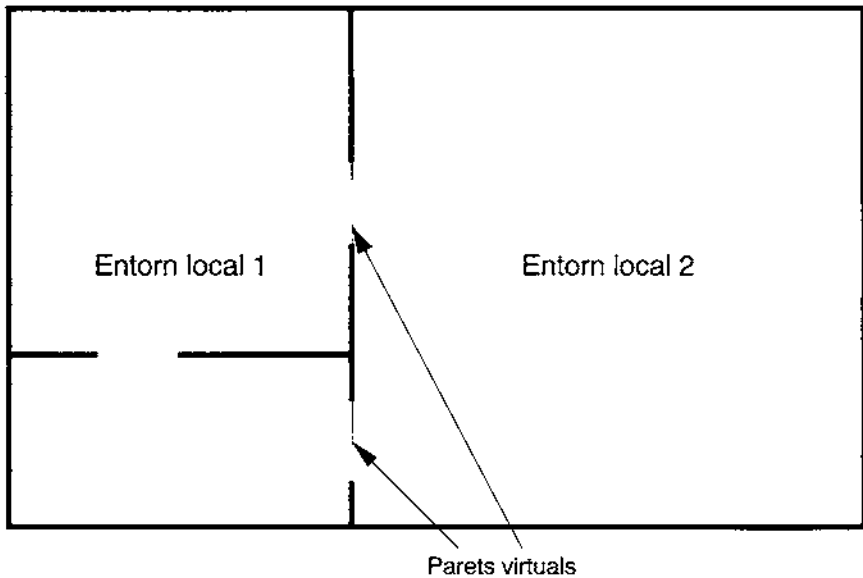


Figura 14. Subdivisió d'una escena utilitzant parets virtuals.

A [Lie 91] es presenta un mètode paral·lel semblant a l'anterior per solucionar la radiositat progressiva, utilitzant una divisió de l'entorn en diverses parts independents que interactuen.

L'algoritme proposat divideix l'escena en un cert nombre de subescenes disjunttes aplicant-hi superfícies fictícies que s'utilitzaran per simplificar l'intercanvi d'energia entre subescenes veïnes. A continuació s'aplica a cada subescena, de manera independent, el mètode de la radiositat progressiva, i se'n fa només una solució parcial. L'algoritme es podria resumir de la manera següent:

```

MENTRE NO Converteix FER
  PER CADA SubEscena FER
    ResoldreRadiositatSubescena
  FPER
  PER CADA Frontera FER
    TransferirEnergia
  FPER
  VisualitzarEscena
FMENTRE
    
```

A [C&P 91] es presenta el mètode de la radiositat progressiva aplicada a un sistema multiprocessador de memòria distribuïda. S'estudien diferents configuracions de connexió. La solució proposada distribueix l'escena entre els diversos processadors, de manera que cada un disposi d'una part equivalent de pedaços. Cada processador determina el seu pedaç més emissor. Com que l'escena està distribuïda, no es pot realitzar tot el procés localment, sinó que cal enviar les dades entre processadors. Les topologies d'interconnexió proposades són: anell, hipercub, malla bidimensional (o torus), malla tridimensional i estructura de xarxa amb distància mínima (AMD, figura 15), en la qual es minimitza el nombre de nodes que cal passar entre dos qualsevol.

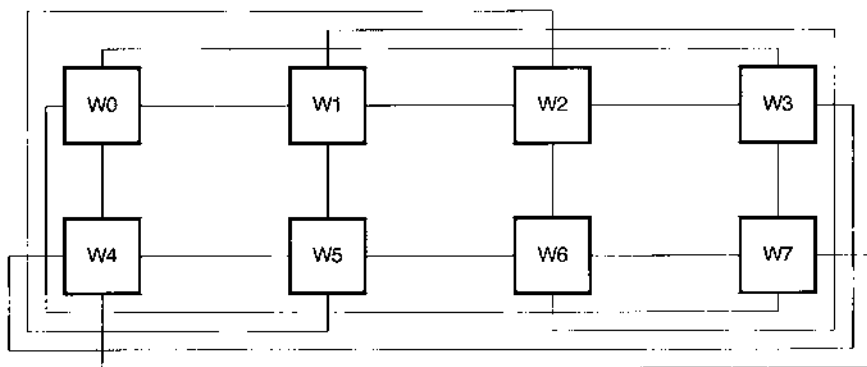


Figura 15. Connexió en AMD de 8 elements.

Una proposta semblant a l'anterior es pot trobar a [F&P 91], on es presenta una implementació paral·lela de l'algorisme de la radiositat progressiva utilitzant també una xarxa de Transputers. La distribució de les dades entre els processadors porta com a conseqüència un intercanvi d'informació entre ells (pas de missatges) per conèixer la part d'escena que no guarden. Amb l'objectiu de minimitzar les comunicacions, els Transputers s'han connectat utilitzant la topologia del camí mínim.

En l'execució de l'algorisme se selecciona l'element més energètic globalment i es realitza el càlcul de factors de forma. Per reduir el nombre de comunicacions necessàries, els pedaços per enviar s'agrupen en blocs d'una certa mida.

Una altra referència que també estudia una paral·lelització semblant és [Vil 92]. Aquí s'estudien diferents versions paral·leles del càlcul de la visibilitat aprofitant la coherència entre pedaços, aplicades a un sistema multiprocessador de memòria distribuïda considerant que l'escena es troba dividida entre els diferents processadors, que cada processador calcula la visibilitat d'un pedaç cap a la resta de l'escena i que per fer-ho requereix saber informació que es troba en altres processadors.

Una primera possibilitat consisteix en el fet que cada node determini la visibilitat completa del pedaç que li pertoca, utilitzant la tècnica de l'hemicub i enviant missatges per sol·licitar els elements que no són locals (figura 16).

Una segona possibilitat de paral·lelisme podria consistir a fer el càlcul parcial de la visibilitat a cada processador i enviar resultats parcials entre els processadors. S'estudien dues variants: que tots els processadors treballin en el mateix hemicub i s'enviïn projeccions parcials o bé fer la projecció completa de tota la base de dades local del processador en l'hemicub i enviar-lo cap al processador següent.

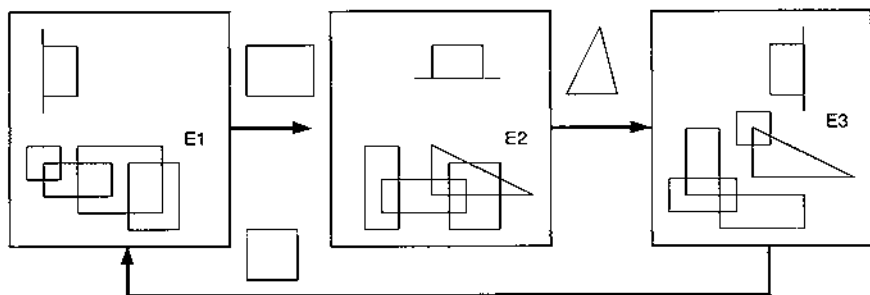


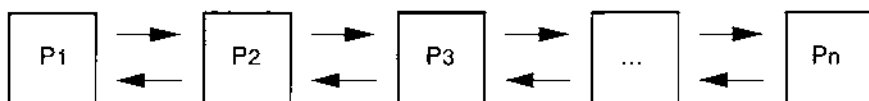
Figura 16. Pedaços enviats a través dels processadors a [Vil 92].

Un altre article que també realitza la paral·lelització amb una xarxa de processadors és [GRS 91], que desenvolupa dos mètodes de càlcul de la radiositat progressiva utilitzant la tècnica del traçat de raigs, un amb la variant determinista i l'altre amb l'estocàstica. Per fer la divisió de les dades es descompon l'escena entre els processadors utilitzant voxels que s'agrupen en llesques, una per processador. La divisió de l'escena en llesques es realitza utilitzant un arbre BSP que també s'utilitza per fer l'equilibri de càrrega estàtic entre els processadors.

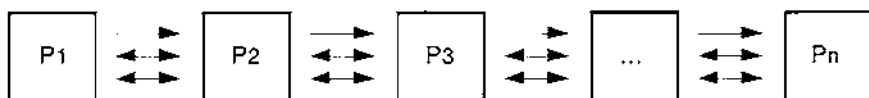
En la radiositat estocàstica (figura 17-a) s'agafa la rajola més energètica (els autors treballen amb rajoles i no amb pedaços, encara que són conceptes semblants) i es tracen un conjunt de raigs de manera aleatòria per difondre la seva energia. Aquest procés es repeteix fins que l'energia que pugui transmetre la rajola que en té més estigui per sota d'un cert llindar. Com que s'ha repartit l'escena entre els diferents processadors, caldrà enviar missatges per resoldre els raigs que no intersectin cap altra rajola dins de les que guarda el processador.

En la implementació determinista (figura 17-b) s'envien les rajoles energètiques entre els processadors i cada un determina la contribució de l'energia que radia la rajola en qüestió sobre les locals (suposant que no hi hagi oclusions). Cada proces-

sador determina l'energia que aporten les rajoles de referència sobre les que té guardades localment.



a) Mètode estocàstic



b) Mètode determinista

Figura 17. Pas de missatges a [GRS 91].

Un enfocament de radiositat en la mateixa línia dels anteriors es pot trobar a [ÇAÓ 93], on es presenta el mètode de la radiositat progressiva utilitzant un sistema multiprocessador de memòria distribuïda i estudiant diferents configuracions de connexió.

La solució proposada distribueix l'escena entre els diversos processadors, de manera que cada un disposi d'una part equivalent de pedaços. Cada processador determina el seu pedaç més emissor i calcula els factors de forma de la resta de l'escena respecte del pedaç de referència. Com que l'escena està distribuïda, cal enviar les dades entre processadors quan es vagin requerint, encara que això pot rebaixar notablement l'eficiència del mètode. També caldrà pensar a distribuir l'energia des dels pedaços de referència cap als pedaços no locals utilitzant missatges.

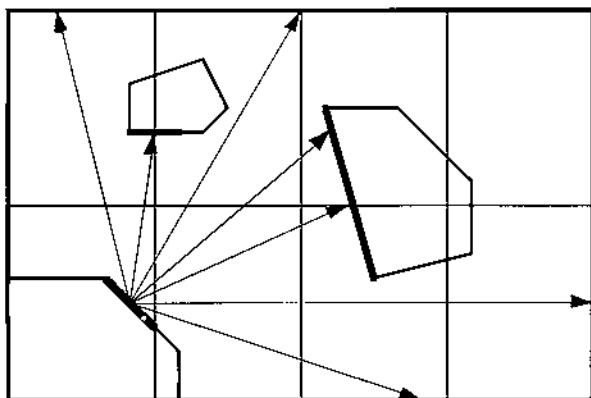


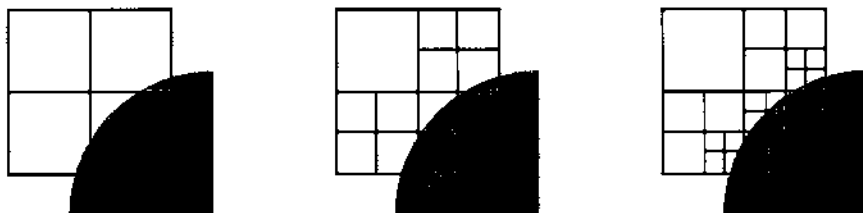
Figura 18. Intersecció de raigs per pas de missatges [GRS 95].

A [GRS 95] es presenta un mètode de paral·lelització del càlcul de radiositat aplicat al cas de radiositat progressiva estocàstica amb un sistema MIMD de memòria distribuïda, dividint l'escena entre els diversos processadors. Aquesta proposta es pot considerar una evolució de [GRS 91]. La divisió de l'escena es fa seguint criteris volumètrics: cada processador té associat un volum de l'escena per tractar i realitza el càlcul de la radiositat progressiva en aquest volum. Quan un procés genera un raig, mira primer la intersecció amb els pedaços locals i, si no en troba cap, l'envia cap a la zona veïna (figura 18).

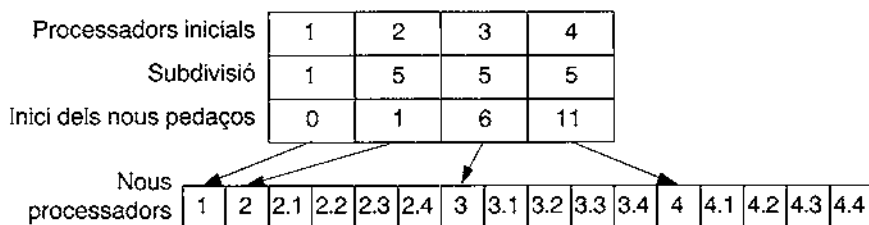
Per millorar l'equilibri de càrrega dels processadors, l'escena es divideix en més parts que processadors existeixen i cada processador agafa un cert nombre de parts diferents.

L'enfocament utilitzat a [D&S 92] és molt diferent dels anteriors. En aquest document es presenta un algoritme de càlcul de la radiositat utilitzant una arquitectura massivament paral·lela de tipus SIMD. Per resoldre la visibilitat amb tècniques paral·leles es proposen dos mètodes: un tracta els objectes de manera successiva i els raigs en paral·lel (*object-serial/ray-parallel*) i un altre utilitza paral·lelisme a tots dos nivells, objectes i raigs (*object-parallel/ray-parallel*). En aquest darrer cas es fan servir processadors virtuals (cada processador físic conté diversos processadors virtuals) que es poden obtenir de manera dinàmica.

La subdivisió adaptativa dels pedaços permet obtenir ombres més nítides en els llocs on hi ha un salt bruscat en la il·luminació. Per fer-ho es divideixen els pedaços de manera adaptativa segons la radiositat en els vèrtexs (figura 19).



a) Subdivisió adaptativa dels pedaços (dos passos)



b) Efecte sobre els processadors (només un pas)

Figura 19. [D&S 92].

Un enfocament relativament semblant es pot trobar a [G&P 95], on es presenta un estudi de diferents mètodes per paral·lelitzar la radiositat progressiva en una arquitectura SIMD. La paral·lelització en un sistema SIMD consisteix a trobar una manera eficient per distribuir les dades en els processadors. El mètode usa tres estructures de dades: els pedaços emissors, els pedaços receptors i els raigs llençats sobre la semiesfera. Com que només hi pot haver una única estructura de dades activa, cal que les altres dues es processin de manera iterativa o desplaçant-les entre els processadors. Com que hi ha tres estructures de dades (emissors, receptors i raigs) i tres possibles maneres de treballar una estructura de dades, es poden trobar 6 possibilitats de treball, encara que el procés en paral·lel dels pedaços emissors no es pot fer de manera massiva, cosa que dóna lloc a només 4 possibilitats (figura 20).

En la primera alternativa es processen en paral·lel tots els raigs que surten des d'un únic pedaç emissor. Cada processador disposa d'una part de l'escena i determina les interseccions del raig traçat amb el conjunt de pedaços de què disposa en aquell moment mentre es desplacen els pedaços entre els processadors. En la segona alternativa es processen en paral·lel les diferents parts de l'escena i es desplacen els raigs que parteixen també d'un únic emissor. En la cinquena alternativa (la tercera i la quarta són les no viables) es processen en paral·lel els pedaços receptors, mentre els elements emissors es desplacen. L'energia s'envia utilitzant un raig comú a tots els processadors. En la sisena alternativa es fa un enfocament igual al del cas anterior, però ara els pedaços emissors es processen en paral·lel i els receptors es desplacen entre els processadors.

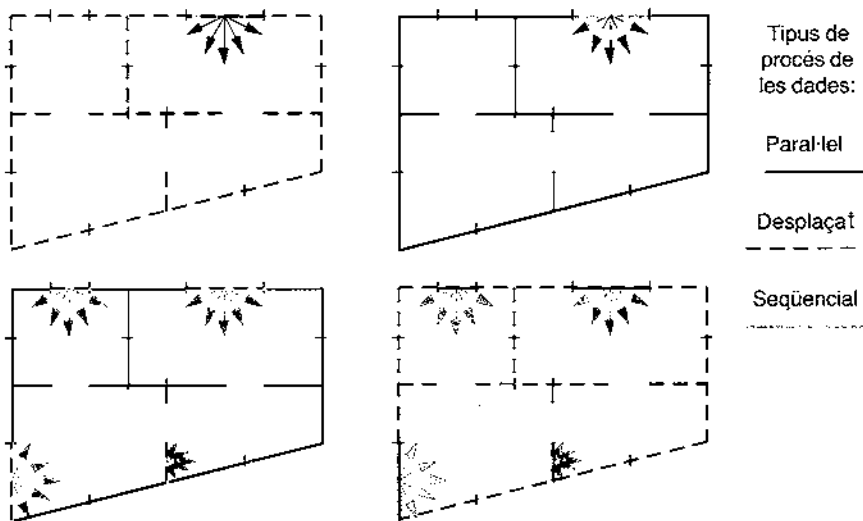


Figura 20. Possibilitats de paral·lelització a [G&P 95].

Una altra implementació massivament paral·lela es pot trobar a [RBL 95]. Es presenta una tècnica per calcular la radiositat utilitzant el mètode progressiu basat en una projecció semiesfèrica, en el qual s'han utilitzat tècniques d'optimització per

reduir el temps de càlcul. L'avantatge de la semiesfera sobre els altres mètodes és la seva homogeneïtat, que permet aconseguir millors resultats. El seu principal inconvenient és el major nombre de càlculs.

A [SSV 95] es presenta un mètode de paral·lelització de la radiositat progressiva sobre un arquitectura MIMD de memòria local, en què s'apliquen diverses tècniques d'equilibri de càrrega.

L'algoritme paral·lel bàsic determina el pedaç més emissor global i s'envia a tots els processadors que fan un càlcul de la visibilitat en paral·lel amb els pedaços que tenen assignats i n'actualitzen la radiositat.

Com a mètodes d'equilibri de càrrega, se'n proposen de tipus estàtic i dinàmic. Per fer equilibri estàtic se suposa que cada element requerirà la mateixa feina i es realitza una distribució equilibrada d'elements. L'equilibri dinàmic s'aconsegueix transferint feina des dels processadors que encara en tenen de pendent cap als processadors que ja l'han acabada.

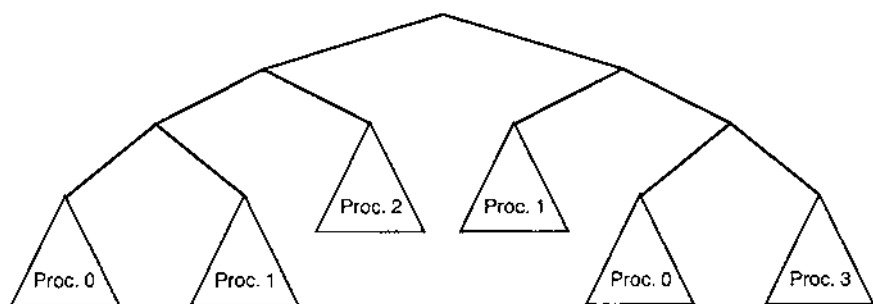


Figura 21. Divisió BSP de l'escena i distribució [SSV 95].

Comparació d'algoritmes

Comparant els diferents algoritmes, es poden destacar els aspectes següents:

- Hi ha algunes tècniques de paral·lelització que treballen amb sistemes SIMD [D&S 92] [G&P 95] [RBL 95], mentre que la resta es basen en sistemes MIMD de memòria distribuïda.
- Totes les referències es basen en tècniques de radiositat progressiva i algunes permeten treballar amb radiositat estesa [Lie 91] [RBL 95].
- Hi ha diverses propostes de mètodes per fer l'equilibri estàtic de càrrega [GRS 91] [D&S 92] [ÇAÓ 93] [G&P 95] [GRS 95] [RBL 95] i una de dinàmica [SSV 95].
- Moltes referències realitzen el càlcul dels factors de forma utilitzant la tècnica de l'hemicub [APV 91] [C&P 91] [F&P 91] [Vil 92] [ÇAÓ 93] o el traçat de raigs [D&S 91] [G&P 95] [SSV 95], si bé també hi ha referències que treballen amb tècniques estocàstiques [GRS 91] [GRS 95].
- La majoria de treballs fan una quantificació del paral·lelisme obtingut (acceleració paral·lela).
- A totes les referències pren molta importància la comunicació d'informació, a causa de la naturalesa dels algoritmes de radiositat.

CONCLUSIONS

En aquest article s'ha presentat un estudi de la paral·lelització dels algoritmes de càlcul de visibilitat, tant en el seu ús en la visualització tridimensional com en les seves aplicacions principals, l'animació i la radiositat. Aquest estudi s'ha centrat principalment en algoritmes que treballen usant tècniques d'escombratge línia a línia.

Un dels aspectes que es pot destacar d'aquest estudi és que la majoria de tècniques de paral·lelització treballen amb sistemes MIMD de memòria distribuïda aplicant tècniques de pas de missatges com a paradigma de programació, tot i que hi ha algunes referències que treballen amb sistemes SIMD (principalment en radiositat) o amb sistemes MIMD de memòria compartida. Per entendre aquesta uniformització d'arquitectures només es poden adduir raons de mercat. Cal esperar que l'evolució actual del mercat cap a sistemes MIMD de memòria compartida o sistemes MIMD de memòria virtual compartida provocarà, en un futur no gaire llunyà, l'aparició de tècniques de paral·lelització que aprofitaran les particularitats d'aquests sistemes.

Generalment, en moltes de les referències, es tracta de mètodes que es basen en la tècnica del *z-buffer* per fer el càlcul de la visibilitat, ja que la tècnica de l'hemicub en radiositat es pot considerar equivalent al *z-buffer*.

Cal destacar que, mentre que en les tècniques d'animació i les de visualització tridimensional es fa el paral·lelisme a l'interior del càlcul de visibilitat (processament en paral·lel de diferents elements), en radiositat sol aplicar-se el paral·lelisme a l'exterior del càlcul de visibilitat (processament en paral·lel de diversos càlculs de visibilitat). Per justificar-ho pot pensar-se en el tipus d'algoritme que es treballa en cada cas: en animació interessa obtenir ràpidament cada una de les imatges independentment de la resta i en radiositat interessa fer ràpidament el càlcul de tots els elements de l'escena.

L'aparició de les tècniques d'equilibri de càrrega és un fet relativament actual dintre del món del paral·lelisme i ha calgut incorporar-lo en els algoritmes quan els càlculs fets han donat lloc a processos molt costosos difícilment distribuïbles de manera equilibrada entre un nombre cada cop més gran de processadors. Dels mètodes presentats en aquest article, els més recents incorporen tècniques d'equilibri de càrrega entre els processadors, ja sigui en forma estàtica (la majoria) o en forma dinàmica (només uns quants). L'ús de mètodes dinàmics d'equilibri de càrrega està molt limitat a problemes extremadament grans, ja que, si no, provoquen un sobre-cost que no en compensa la utilització.

També es pot observar una importància cada cop més gran de les tècniques d'avaluació del paral·lelisme: no n'hi ha prou de paral·lelitzar, cal mesurar els resultats obtinguts. Tanmateix, aquest és un aspecte molt poc treballat, ja que els resultats que es donen són difícilment comparables atès que es mesuren de manera empírica utilitzant escenes diferents.

Finalment, es pot destacar que hi ha tècniques de paral·lelització molt diverses, però totes tenen limitacions pel que fa a l'escalabilitat (increment del nombre de processadors), causades principalment per les comunicacions i els desequilibris de càrrega entre els processadors. Generalment no sol haver-hi limitacions pel que fa a memòria del sistema, probablement a causa del constant abaratiment de preus que experimenta.

AGRAÏMENTS

Aquest estudi s'ha realitzat dins el marc del projecte TIC95-0639-C05-05 finançat per la CICYT. Voldria agrair el suport i els comentaris aportats pel Dr. X. Pueyo i el Dr. J. Regincós durant el desenvolupament de tot aquest treball.

Bibliografia

- [APV 91] ARNALDI, Bruno; PUEYO, Xavier; VILAPLANA, Josep. **On the division of environments by virtual walls for radiosity computation.** *Eurographics Workshop on Rendering*. 1991, 2, 1-10.
- [Bou 70] BOUKNIGHT, W.Jack. **A procedure for generation of three-dimensional half-toned computer graphics presentations.** *Communications of the ACM*, 1970, 13(9), 527-536.
- [C&O 93] CROCKETT, Thomas W.; ORLOFF, Tobias. **A MIMD rendering algorithm for distributed memory architectures.** *Proceedings of the 1993 Rendering Symposium*. 1993, 35-42.
- [C&P 91] CHALMERS, Alan G.; PADDON, Derck J. **Parallel processing of progressive refinement radiosity methods.** *Eurographics Workshop on Rendering*, 1991, 2, 1-14.
- [C&R 81] CHANG, P.; RAMESH, Jain. **A multi-processor system for hidden-surface-removal.** *Computer Graphics*, 1981, 14(4), 405-436.
- [ÇAÖ 93] ÇAPIN, Tolga K.; AYKANAT, Cevdet; ÖZGÜÇ, Bülent. **Progressive refinement radiosity on ring-connected multicomputers.** *Proceedings of the 1993 Parallel Rendering Symposium*. 1993, 71-76.
- [CCW 88] COHEN, Michael F.; CHIEN, Shenchang Eric; WALLACE, John R.; GREENBERG, Donald P. **A progressive refinement approach to fast radiosity generation.** *Computer Graphics*. 1988, 22 (4), 75-84.
- [D&S 92] DRUCKER, Steven M.; SCHRÖDER, Peter. **Fast radiosity using a data parallel architecture.** *Eurographics Workshop on Rendering*, 1992, 3(-), 247-258.
- [D&W 87] DYER, Scott; WHITMAN, Scott. **A vectorized scan-line z-buffer rendering algorithm.** *IEEE Computer Graphics & Applications*, 1987, 7(7), 34-45.
- [Eil 93] ELLSWORTH, David. **A multicomputer polygon rendering algorithm for interactive applications.** *Proceedings of the 1993 Parallel Rendering Symposium*. 1993, 43-48.
- [Eil 94] ELLSWORTH, David. **A new algorithm for interactive graphics on multicomputers.** *IEEE Computer Graphics and Applications*. 1994, 14 (4), 33-40.
- [F&P 91] FEDA, Martin; PURGATHOFER, Werner. **Progressive refinement radiosity on a transputer network.** *Eurographics Workshop on Rendering*. 1991, 2, 1-10.
- [FFR 83] FIUME, Eugene; FOURNIER, Alain; RUDOLPH, Larry. **A parallel scan conversion algorithm with anti-aliasing for a general-purpose ultracomputer.** *Computer Graphics*, 1983, 17(2), 141-150.
- [Fly 66] FLYNN, Michael J. **Very high-speed computing systems.** *Proceedings of the IEEE*, 1966, 54(12), 1901-1909.
- [G&P 95] GARCÍA, Germán; PUEYO, Xavier. **Progressive radiosity solutions on SIMD architectures.** *Congreso Español de Informática Gráfica*. 1995, 5, 339-352.

- [GGH 88] GHARACHORLOO, Nader; GUPTA, Satish; HOKENEK, Erdem; BALASUBRAMANIAN, Peruvemba; BOGHOLTZ, Bill; MATHIEU, Christian; ZOULAS, Christos. **Subnanosecond pixel rendering with a million transistor chip.** *Computer Graphics*, 1988, 22(4), 41-49.
- [GRS 91] GUITTON, P.; ROMAN, J.; SCHLICK, C. **Two parallel approaches for progressive radiosity.** *Eurographics Workshop on Rendering*, 1991, 2, 1-15.
- [GRS 95] GUITTON, Pascal; ROMAN, Jean; SUBRENAT, Gilles. **Implementation results and analysis of a parallel progressive radiosity.** *Proceedings of the 1995 Parallel Rendering Symposium*. 1995, 31-38.
- [GTG 84] GORAL, C. M.; TORRANCE, K. E.; GREENBERG, D. P.; BATAILE, B. **Modelling the interaction of lights between diffuse surfaces.** *Computer Graphics*. 1984, 18 (3), 212-222.
- [H&B 92] HIGHFIELD, Julian C.; BEZ, Helmut E. **Hidden surface elimination on parallel processors.** *Computer Graphics Forum*, 1992, 11(5), 293-307.
- [Hoa 78] HOARE, C.A.R. **Communicating sequential processes.** *Communications of the ACM*, 1978, 21(8), 666-677.
- [HSA 91] HANRAHAN, Pat; SALZMAN, David; AUPPERLE, Larry. **A rapid hierarchical radiosity algorithm.** *Computer Graphics*. 1991, 25 (4), 197-206.
- [IAB 91] ISLER, Veysi; AYKANAT, Cevdet; BÜLEN, Özgüç. **Subdivision of 3D space based on the graph partitioning for parallel ray tracing.** *Eurographics Workshop on Rendering*, 1991, 2, 1-13.
- [K&G 79] KAPLAN, Michael; GREENBERG, Donald P. **Parallel processing techniques for hidden surface removal.** *Computer Graphics*, 1979, 13(7), 300-307.
- [Lie 91] Van LIÈRE, R. **Divide and conquer radiosity.** *Eurographics Workshop on Rendering*. 1991, 2, 1-11.
- [LRN 95] LEE, Tong-Yee; RAGHAVENDRA, C.S.; NICHOLAS, J. N. **Image composition methods for sort-last polygon rendering on 2D mesh architectures.** *Proceedings of the 1995 Parallel Rendering Symposium*. 1995.
- [Luc 90] LUCAS, Michael. **Parallélisme et synthèse d'image.** 1990.
- [MCE 94] MOLNAR, Steven; COX, Michael; ELLSWORTH, David; FUCHS, Henry. **A sorting classification on parallel rendering.** *IEEE Computer Graphics and Applications*. 1994, 14 (4), 23-32.
- [Mue 95] MUELLER, Carl. **The sort-first rendering architecture for high-performance graphics.** *Proceedings of the 1995 Symposium on Interactive 3D Graphics*. 1995, 29 (2), 75-84.
- [Pue 86] PUEYO, Xavier. **Estudi dels algorismes de visualització d'escenes tridimensionals formades per superfícies corbes i mixtes utilitzant el principi d'escombrat línia a línia.** *Tesi doctoral (Ph. D.), Universitat Politècnica de Catalunya*, 1986.
- [Pue 91] PUEYO, Xavier. **Diffuse interreflexions. Techniques for form-factor computation: a survey.** *The Visual Computer*, 1991, (7), 200-209.
- [RBL 95] RENAUD, C.; BRICOUT, F.; LEPRÊTRE, E. **Massively parallel hemispherical projection for progressive radiosity.** *Computer & Graphics*. 1995, 19 (2), 273-279.
- [Rob 88] ROBLE, D. R. **A load balanced parallel scanline z-buffer algorithm for the iPSC hypercube.** *Proceedings of Pixim*, 1988, 177-192.

- [S&P 89] SILLION, François; PUECH, Claude. **A general two-pass method integrating specular and diffuse reflection.** *Computer Graphics*. 1989, 23 (3), 335-344.
- [SPG 93] SCOPIGNO, R.; PAOLUZZI, A.; GUERRINI, S.; RUMOLO, G. **Parallel depth-merge: a paradigm for hidden surface removal.** *Computer & Graphics*. 1993, 17 (5), 583-592.
- [SSS 74] SUTHERLAND, Ivan E.; SPROULL, Robert F.; SCHUMACKER, Robert A. **A characterization of ten hidden-surface algorithms.** *Computing Surveys*, 1974, 6(1).
- [SSV 95] STÜRZLINGER, W.; SCHAUFLER, G.; VOLKERT, J. **Load balancing for a parallel radiosity algorithm.** *Proceedings of the 1995 Parallel Rendering Symposium*. 1995.
- [The 87] THEOHARIS, Theoharis. **Algorithms for parallel polygon rendering.** *Lecture notes in computer science*, Springer Verlag, 1987.
- [Vil 92] VILAPLANA, Josep. **Parallel radiosity solutions based on partial result messages.** *Eurographics Workshop on Rendering*, 1992, 259-270.
- [Whi 80] WHITTED, Turner. **An improved illumination model for shaded display.** *Communications of the ACM*. 1980, 23 (6), 343-349.
- [Whi 92] WHITMAN, Scott. **Parallel graphics rendering algorithms.** *Eurographics Workshop on Rendering*, 1992, 3, 123-134.
- [Whi 93] WHITMAN, Scott. **A task adaptative parallel graphics renderer.** *Proceedings of the 1993 Parallel Rendering Symposium*. 1993, 27-34.
- [Whi 94] WHITMAN, Scott. **Dynamic load balancing for parallel polygon rendering.** *IEEE Computer Graphics and Applications*. 1994, 14 (4), 41-48.