



**HAL**  
open science

## Subdivision spatiale et lancer de rayon

Bruno Arnaldi, Thierry Priol, Kadi Bouatouch

► **To cite this version:**

Bruno Arnaldi, Thierry Priol, Kadi Bouatouch. Subdivision spatiale et lancer de rayon. [Rapport de recherche] RR-0589, INRIA. 1986. inria-00075965

**HAL Id: inria-00075965**

**<https://inria.hal.science/inria-00075965>**

Submitted on 24 May 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# IRIA

CENTRE DE RENNES

IRISA

Institut National  
de Recherche  
en Informatique  
et en Automatique

Domaine de Voluceau  
Rocquencourt  
BP 105

78153 Le Chesnay Cedex  
France

tel (1) 39 63 55 11

## Rapports de Recherche

N° 589

### SUBDIVISION SPATIALE ET LANCER DE RAYON

Bruno ARNALDI  
Thierry PRIOL  
Kadi BOUATOUCH

Décembre 1986

Campus Universitaire de Beaulieu  
Avenue du Général Leclerc  
35042 - RENNES CÉDEX  
FRANCE  
Tél. : (99) 36.20.00  
Télex : UNIRISA 95 0473 F

## SUBDIVISION SPATIALE ET LANCER DE RAYON

Bruno ARNALDI, Thierry PRIOL, Kadi BOUATOUCH

### SPACE SUBDIVISION AND RAY-TRACING

Publication Interne n° 322 - Novembre 86

12 pages

**Résumé** : Nous décrivons dans ce rapport, deux algorithmes de subdivision spatiale pour la méthode de lancer de rayon. Ces algorithmes permettent des facteurs d'accélération de 10 à 40 suivant la complexité des images à synthétiser. La première méthode utilise un découpage irrégulier de la scène 3D permettant d'épouser au mieux les formes des objets. Le nombre d'intersection entre un rayon et les primitives géométriques est réduit au maximum. La deuxième méthode consiste à découper la scène 3D de façon régulière. La structure de donnée ainsi créée est plus petite que celle générée par la première méthode mais elle est moins rapide dans l'étape de lancer de rayons.

**Abstract** : We describe in this report, two space subdivision algorithms for ray-tracing method. These algorithms permit to reduce time in a factor of 10 to 40 depending of the picture complexity. The first method uses an irregular space subdivision which permits to save a great number of intersection between a ray and elementary primitives. The second method divides space in a regular fashion. The data base created by this method takes less space in memory than the first one but it is slower in the ray-tracing step.

## 1. Introduction.

La méthode de synthèse d'image par lancer de rayon est incontestablement celle qui produit les images les plus réalistes. Elle présente par contre l'inconvénient d'être coûteuse en temps CPU. Trois techniques ont été proposées pour remédier à cet inconvénient :

La première consiste à tirer profit de la cohérence image (pixels) [BRONSVOORT 84] en calculant l'intensité de macropixels. L'intensité des autres pixels est déduite par extrapolation, en fonction de l'intensité mesurée des pixels voisins. Cette technique présente l'avantage de réduire le nombre de rayons primaires et le désavantage de faire disparaître des tranche d'images, ce qui va à l'encontre de l'objectif de la méthode du lancer de rayon qui est la production d'images d'un haut niveau de réalisme.

La deuxième technique [COQUILLART 85] consiste à exploiter la cohérence objet, par projection des volumes englobants des objets de la scène sur un plan perpendiculaire à l'un des axes de coordonnées de l'espace image. La structure de données contient les relations d'adjacence et de superposition des volumes englobant projetés. Cette méthode de projection sur un plan ne permet pas de tirer profit de la cohérence spatiale de façon optimum.

La troisième technique consiste à subdiviser l'espace 3D contenant la scène en régions parallélépipédiques, ne contenant qu'un nombre minimum de primitives géométriques. Un rayon primaire ou secondaire pénétrant dans une région n'intersectera que les primitives contenues dans cette région. Cette technique est de loin celle qui permet de réduire le plus possible le temps de synthèse, qui devient alors relativement indépendant du nombre de primitives géométriques de la scène.

Plusieurs auteurs ont proposé des méthodes de subdivision spatiale (troisième technique). Ils ont montré que ces types d'algorithmes améliorait notablement l'efficacité du lancer de rayon pour des scènes complexes. Cependant, elles nous conduisent à formuler les remarques suivantes.

- L'utilisation d'une structure de donnée, organisée en octree et d'une table d'indexage, décrit par Glassner [GLASSNER 84], rend très coûteux la recherche de la région suivante dans la direction d'un rayon. En effet, cette recherche consiste à parcourir tout l'arbre octree et d'effectuer des comparaisons avec les faces des voxels. Le type de modélisation utilisé n'est pas précisé.

- L'organisation de la structure de donnée en arbre binaire de type BSP proposée par Kaplan [KAPLAN 85] , nécessite la traversée de tout cet arbre pour déterminer les régions se trouvant dans la direction d'un rayon. Si la modélisation est de type CSG, Kaplan propose de calculer l'intersection entre un rayon et toute les primitives se trouvant dans toutes les régions situées sur le trajet du rayon. Ce calcul s'avère très coûteux.

- Fujimoto et al [FUJIMOTO 86] utilisent une structure de donnée de type énumération spatiale appelée SEADS et un analyseur différentiel : le 3DDA. L'application de cette structure à une de modélisation de type CSG n'est pas envisagée dans leur article.

Dans tous les articles traitant de la subdivision spatiale, l'aspect modélisation est peut explicité. Nous proposons, dans ce texte, une modélisation simple et exacte permettant de définir une grande variété d'objets: l'arbre CSG.

Pour remédier aux inconvénients des méthodes précédentes, nous avons étudié et mis en oeuvre deux techniques de subdivision :

#### **Subdivision spatiale par découpage irrégulier.**

Le découpage spatial s'effectue de façon irrégulière ; il est lié à la disposition des objets dans l'espace. Les régions résultant du découpage "collent" bien à la forme des objets. Le temps de synthèse est alors minimisé, mais le passage d'une région à une autre a nécessité la mise en oeuvre d'algorithmes spécialisés.

#### **Subdivision spatiale régulière.**

Le découpage est dans ce cas régulier ; l'intérêt étant alors d'améliorer la rapidité des accès aux informations : passage d'une région à une autre.

## 2. Subdivision spatiale par découpage irrégulier en modélisation CSG.

Les techniques classiques de subdivision spatiales offrent un découpage régulier de l'espace de la scène. Parmi ces méthodes nous pouvons citer celle de Glassner, utilisant une subdivision par octree, et celle de Kaplan, effectuant un BSP (Binary Space Partitioning). Malgré la possibilité d'interrompre la subdivision d'une boîte selon certains critères d'arrêts, ces techniques ne permettent pas de "coller", de façon optimale à la répartition spatiale des objets de la scène. Par contre, la structure de donnée obtenue permet de suivre facilement le trajet d'un rayon dans la scène, afin d'identifier toutes les sous-régions rencontrées.

Nous proposons une approche de la subdivision spatiale tridimensionnelle fondée sur un découpage lié à la disposition des objets dans l'espace de la scène [ARNALDI 86]. Contrairement aux techniques classiques, il en résulte une répartition irrégulière de l'ensemble des sous-régions.

La subdivision spatiale est un prétraitement de l'étape de synthèse d'une image. Pour simplifier cette étape, tout en respectant la disposition des objets, chaque primitive est remplacée par son volume englobant parallélépipédique, que l'on minimise en fonction des opérateurs de l'arbre CSG initial (Figure 1).

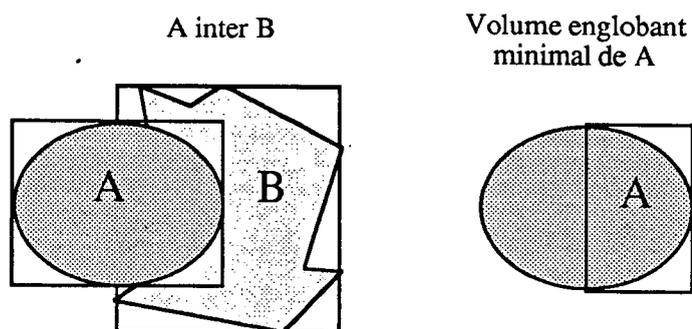


Figure 1 : Volume englobant minimal.

Le découpage spatial est alors réalisé en deux étapes: 2D et 3D.

a). 2 dimensions :

Le plan infini dans lequel se situe l'écran virtuel est choisi comme plan privilégié pour une subdivision en deux dimensions. Chaque volume englobant est projeté sur ce plan, selon la position de l'observateur virtuel, de façon à obtenir un rectangle que l'on décompose en ses quatre segments de droite. Cet ensemble de segments est alors utilisé pour réaliser un partitionnement de l'écran (Figure 2) par la technique du BSP décrite par Fuchs [FUCHS 80]. On choisit un segment dans l'ensemble des segments disponibles. La droite extension de ce segment réalise une partition binaire du plan. L'ensemble des segments restant est réparti entre les deux régions ainsi créées auxquelles on applique récursivement le même traitement, et ce jusqu'à épuisement des segments. Chaque partie du plan peut être vide ou posséder un certain nombre de primitives.

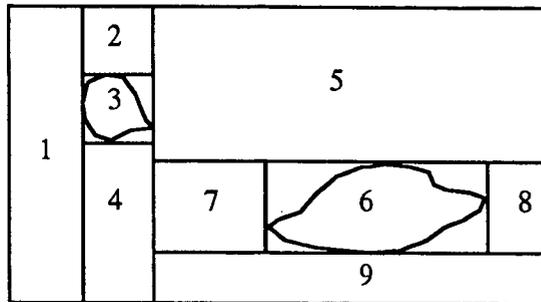


Figure 2 : Découpage de l'écran en 9 parties.

b). 3 dimensions :

Chacune de ces zones non vides est à son tour découpée dans la troisième dimension par l'utilisation des valeurs englobantes sur cet axe. Comme le montre la figure 3, l'extension 3D de la partie d'écran est découpée selon l'axe 'z' suivant la répartition des primitives. En effet, cette région possède, après l'étape 1, deux primitives A et B, avec les valeurs minimales et maximales :  $z_{minA}$ ,  $z_{minB}$ ,  $z_{maxA}$  et  $z_{maxB}$ . Ces valeurs permettent de partitionner la région 3D en sous régions contenant ou non les primitives.

Ecran

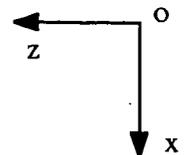
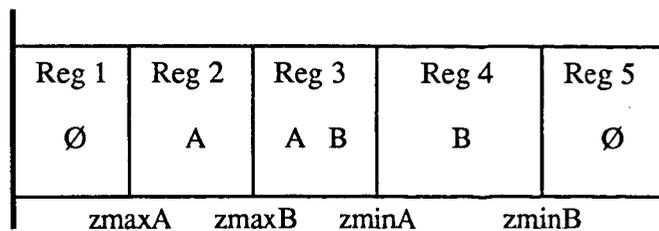


Figure 3 : Découpage en profondeur.

Le problème du passage d'une région à l'autre, pour suivre le trajet d'un rayon dans la scène, est résolu par l'ajout d'informations de connexité pour chaque région de la base. Ces informations représentent, pour une boîte, la connaissance de son environnement, et sont exprimées par des pointeurs (Figure 4) liés à certains coins de la région. Un pointeur de coin permet de passer directement d'une région à la région qui lui est adjacente par le coin. Cette structure de pointeurs est suffisante pour garantir le passage de n'importe quelle région de la base, à n'importe quelle autre région.

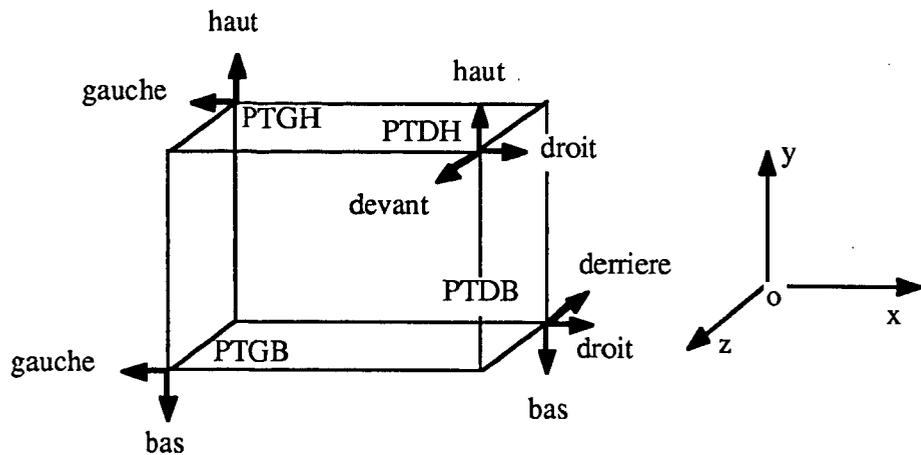


Figure 4 : Structure de pointeurs pour la connexité.

Un exemple en deux dimensions (figure 5) montre comment il est possible de suivre le trajet d'un rayon, à travers une structure irrégulière de boîte, en utilisant uniquement ces pointeurs. Les traits fins symbolisent les pointeurs, les flèches indiquent les pointeurs effectivement utilisés dans l'exemple, les ronds noirs montrent les régions traversées par le rayon.

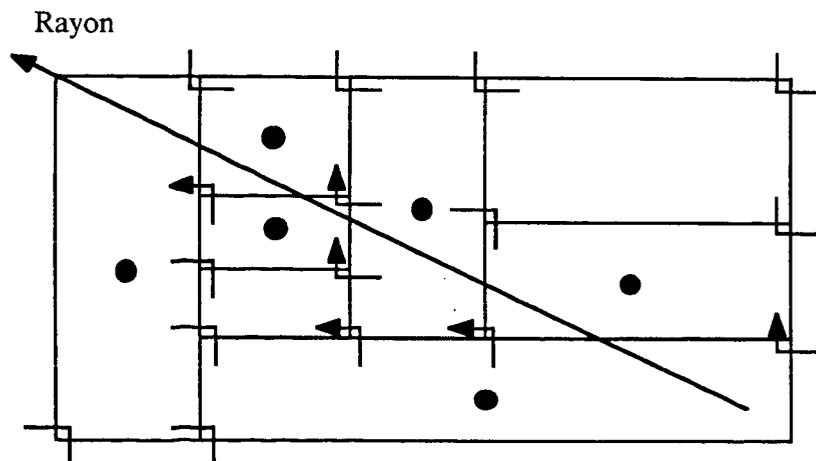


Figure 5 : Suivre le trajet d'un rayon.

La modélisation utilisée est de type CSG. A partir d'un arbre binaire classique, en fonction des primitives effectives de chaque boîte, l'utilisation de quelques règles de construction simples permet d'effectuer la restriction de l'arbre initial, relativement aux primitives de la boîte considérée. La méthode consiste à parcourir l'arbre initial et de reconstruire au fur et à mesure de la remontée de cet arbre la restriction. Pour un noeud donné N, si l'on appelle A et B les restrictions de ses deux sous arbres et op l'opérateur booléen associé, on obtient :

- a). N est une primitive :
- si N appartient à la région
  - alors restriction(N) = N
  - sinon restriction(N) =  $\emptyset$
- b). N est un noeud :
- si A et B sont non vides
  - alors restriction(N) = A op B
  
  - si A et B sont vides
  - alors restriction(N) =  $\emptyset$
  
  - si A est vide et B non vide
  - alors si op=union
  - alors restriction(N) = B
  - si op=difference ou op=intersection
  - alors restriction(N) =  $\emptyset$
  
  - si A est non vide et B vide
  - alors si op=union ou op=difference
  - alors restriction(N) = A
  - si op=intersection
  - alors restriction(N) =  $\emptyset$

Un défaut connu des techniques de subdivision réside dans le fait qu'une primitive peut être répartie entre plusieurs régions de la structure finale. Lors des tests d'intersection, on peut être amené à calculer plusieurs fois l'intersection effective entre un même rayon et une primitive, avant de trouver une valeur compatible avec les bornes inférieures et supérieures de la région courante (Figure 6). Une solution simple pour remédier à ce défaut consiste à associer à chaque primitive une structure de donnée supplémentaire appelée "boîte aux lettres", et de numéroter chaque rayon de manière unique.

A chaque calcul d'intersection effectif entre un rayon et une primitive, on dépose, dans la boîte aux lettres, le numéro du rayon ainsi que les valeurs d'intersection calculées. Lors d'une demande d'intersection ultérieure avec cette primitive, si le numéro du rayon, ayant engendré cette requête, est le même que celui stocké dans la boîte aux lettres, on récupère uniquement les valeurs d'intersection sans effectuer de calcul. Dans le cas contraire, le rayon est différent et le calcul est à faire. Il sera, à son tour déposé dans la boîte aux lettres.

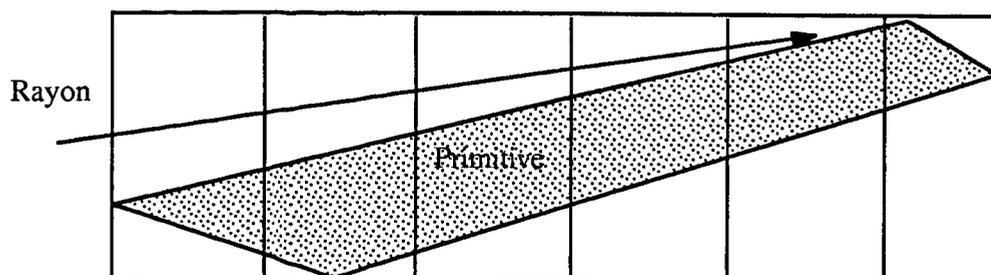


Figure 6 : Cas de plusieurs intersections possibles

La structure de donnée résultant du prétraitement est alors une collection de zones 3D liées par des pointeurs et contenant, ou non, de petits arbres binaires. Elle offre l'avantage d'être directement liée à la répartition des objets dans la scène. Le passage d'une région à l'autre s'effectue très rapidement par l'usage des pointeurs de coin, même si cette structure est irrégulière.

### 3. Accélération des accès aux données pour un découpage spatial régulier

Glassner utilise comme structure d'accès aux données, l'arbre octal ayant servi au découpage spatial. De même, Kaplan utilise l'arbre binaire du BSP pour sélectionner les régions. Chaque passage d'une région à une autre nécessite la mise en oeuvre d'une recherche dans ces arbres. Cette opération est effectuée un tel nombre de fois, dans le processus de synthèse, qu'il nous est apparu indispensable de l'optimiser.

Le découpage proprement dit est réalisé par un algorithme de type BSP, l'espace est subdivisé successivement suivant les trois axes et peut être contrôlé par des paramètres servant à évaluer des critères d'arrêts (niveau maximum de subdivision, nombre maximal d'objet par région). Chaque boîte résultant du découpage est numérotée. L'arbre n'est pas conservé comme structure d'accès aux données, du fait de la lenteur des algorithmes de parcours d'arbre.

L'accès va s'effectuer à travers une table de référence (Directory), la méthode utilisée est inspirée de Tamminen et al [TAMMINEN 82 et 84] qui l'ont employée pour la modélisation de solides par frontières (approchés par des facettes planes). Cette table est un tableau à trois dimensions (x,y,z), dont chaque case contient un numéro de boîte de la structure de donnée. La structure des numéros de la table reflète la structure du découpage spatial. La case élémentaire de cette table de référence correspond, du point de vue géométrique, au niveau de subdivision maximal de l'espace de la scène. Si la boîte n'a pas atteint le niveau maximal de subdivision, plusieurs cases de la table des références porteront son numéro (Figure 7). La construction de cette table est donc directement lié à la subdivision spatiale.

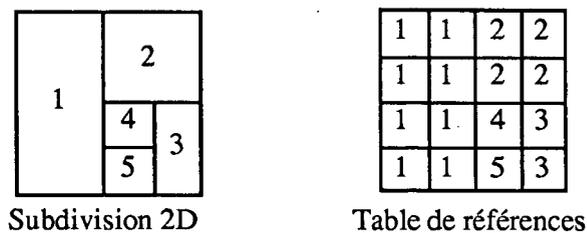


Figure 7 : Exemple 2D de la table de références.

Pour la recherche de la boîte contenant un point P, défini par ses coordonnées (x,y,z), les indices i,j et k de la table de référence, permettant d'accéder à cette boîte, sont donnés par :

$$i = \text{partie entière} \left( \frac{x - vx_{\min}}{vx_{\max} - vx_{\min}} * 2^{\text{niv}_{x\max}} \right) + 1$$

$$j = \text{partie entière} \left( \frac{y - vy_{\min}}{vy_{\max} - vy_{\min}} * 2^{\text{niv}_{y\max}} \right) + 1$$

$$k = \text{partie entière} \left( \frac{z - vz_{\min}}{vz_{\max} - vz_{\min}} * 2^{\text{niv}_{z\max}} \right) + 1$$

où -  $vx_{\min}, vx_{\max}, vy_{\min}, vy_{\max}, vz_{\min}$  et  $vz_{\max}$  délimitent l'espace de scène.

-  $\text{niv}_{x\max}, \text{niv}_{y\max}$  et  $\text{niv}_{z\max}$  sont les niveau de découpage maximum respectivement sur les axes x,y et z

Le contenu de la table  $t[i,j,k]$  fournit le numéro de la boîte et permet donc l'accès aux informations géométriques qu'elle contient.

Pour la recherche de la boîte suivante, sur le trajet d'un rayon, on procède de la façon suivante:

- a). On recherche, dans la boîte courante, les coordonnées du point de sortie du rayon, c'est à dire le point appartenant au rayon situé sur une face de sortie de la boîte.
- b). Si le point se trouve sur une face perpendiculaire à  $x$ ,  $y$  ou  $z$ , on ajoute ou on retranche à la coordonnée en  $x$ ,  $y$  ou  $z$  du point de sortie une valeur  $\text{deltax}$ ,  $\text{deltay}$  ou  $\text{deltaz}$ . Ces valeurs sont ajoutées ou retranchées suivant le signe de  $dx, dy, dz$ , exprimant la direction du rayon. Si le rayon sort par une arête, on "pousse" le point de sortie selon les deux axes perpendiculaires aux faces adjacentes de cette arête. Si le rayon sort par un coin, il en sera de même suivant les trois axes.
- c). Les coordonnées du point résultant vont permettre l'évaluation des indices de la table de référence en utilisant la méthode de recherche d'une boîte contenant un point  $P$  décrite précédemment.

L'intérêt de cette méthode d'accès aux données réside dans le fait que la recherche de la boîte suivante sur le trajet d'un rayon est très rapide et d'un coût de calcul constant. Il est possible de minimiser la taille mémoire occupée par la table de références en introduisant la notion de sous table proposée par Tamminen. La modélisation utilisée est de type CSG, elle est donc exacte et très facile à mettre en oeuvre. Comme dans le cas de la subdivision irrégulière, on associe à une boîte un petit arbre CSG qui est la restriction de l'arbre CSG de toute la scène aux primitives contenues dans cette boîte. Le temps de prétraitement pour une image est très court et de l'ordre de quelques dizaines de secondes CPU sur un VAX 11/750.

## Bibliographie

- [ARNALDI 86] B. ARNALDI, T. PRIOL,  
"Synthèse d'image par lancer de rayon, Subdivision spatiale, Algorithmes et Architecture"  
Rapport de DEA 1986, IFSIC
- [BRONSVOORT 84] F. BRONSVOORT, J.J VAN WIJK and F.W JANSEN,  
"Two method for improving the efficiency of ray casting in solid modelling,"  
*CAD* ,16-1 Jan 1984, pp 51-55.
- [COQUILLART 85] S. COQUILLART,  
"An improvement of the ray tracing algorithm,"  
*in Proc. Eurographics* ,1985, pp 77-88.
- [FUCHS 80] H. FUCHS, Z.M. REDEM and B.F. NAYLOR,  
"On visible surface generation by a priori tree structure,"  
*ACM* , Apr1980, pp 124-133.
- [FUJIMOTO 86] A. FUJIMOTO, T. TANAKA and K. IAWATA,  
"ARTS : Accelerated Ray-Tracing System,"  
*IEEE CG&A* , Apr 1986, pp 16-26.
- [GLASSNER 84] A. GLASSNER,  
"Space subdivision for fast ray tracing,"  
*IEEE CG&A* , Oct 1984, pp 15-22.
- [KAPLAN 85] M.R. KAPLAN,  
"Space tracing, a constant time ray tracer,"  
*in Proc. SIGGRAPH 85*, tutorial on the use of spatial coherence in ray tracing.
- [TAMMINEN 82] M.TAMMINEN and R.SULONEN,  
"The excell method for efficient geometric acces to data,"  
*in Proc. IEEE*, 19th design automation conference 1982,  
Paper 23.1, pp 345-351.
- [TAMMINEN 84] M. TAMMINEN, O. KORONEN and M. MANTYLA,  
"Ray-casting and block model conversion using a spatial index,"  
*CAD* , 16-4 Jul 1984, pp 203-208.

