Algorithme incrémental de mise à jour du maillage de discontinuité pour les environnements dynamiques

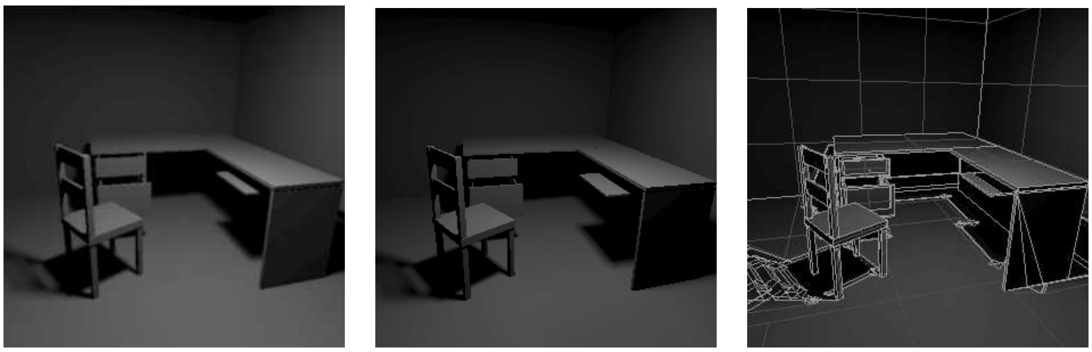
Céline Loscos et George Drettakis
iMAGIS[1]*, Laboratoire GRAVIR/IMAG-INRIA, BP 53, Grenoble Cedex 9, FRANCE.*
{Celine.Loscos|George.Drettakis}@imag.fr

Résumé : *Nous nous intéressons au calcul de l'éclairage d'une scène à partir d'un maillage de discontinuité. La scène est composée de polygones et munie d'une source étendue et nous ne considérons que l'éclairage direct. Le temps de calcul de l'éclairage d'une scène comportant une centaine de polygones exclue l'interactivité avec un utilisateur si les méthodes précédentes sont utilisées. Nous proposons deux méthodes de mise à jour du maillage de discontinuité et de l'éclairage direct permettant l'interaction lors du déplacement d'un objet dynamique présélectionné. La première est basée sur la cohérence spatiale autour du mouvement. Une structure spatiale permet de stocker et de localiser les informations nécessaires aux modifications de l'éclairage. La deuxième exploite la cohérence de visibilité. Elle utilise la connaissance des intersections des surfaces de discontinuité avec les polygones de la scène. Ces deux méthodes ne modifient que la région de la scène dont l'éclairage est affecté par le mouvement de l'objet. L'éclairage ainsi obtenu est le même que celui obtenu avec le recalcul total de la scène. Les algorithmes décrits pourront par la suite être appliqués pour des scènes dont l'éclairage global a été calculé par la méthode de radiosité à partir d'un maillage de discontinuité.*

Mots-clés : éclairage direct, radiosité, maillage de discontinuité, environnements dynamiques, interactivité.

1 Introduction

Nous considérons le calcul de l'éclairage direct d'une scène virtuelle 3D, éclairée par une source étendue. Ce calcul est réalisé en utilisant un maillage de discontinuité [5, 10, 9, 15], qui délimite les zones d'ombre et de pénombre. Ce maillage est une partition des polygones de la scène telle que la partie visible de la source ait la même topologie dans chaque facette du maillage. Cette topologie est représentée dans une structure de données appelée *projection inverse* (en anglais « backprojection » [5, 15]). Le maillage de discontinuité permet d'avoir un éclairage direct de très haute qualité. L'association d'un tel maillage avec un maillage quadtree peut être utilisé pour calculer un éclairage global de haute qualité par la méthode de radiosité hiérarchique [6].



(a) (b) (c)

FIG. 1 - *(a) Scène 3 (b) Scène 3 avec mouvement de la tablette (c) Maillage de la scène 3*

1. iMAGIS est un projet de recherche commun CNRS/INRIA/UJF/INPG.

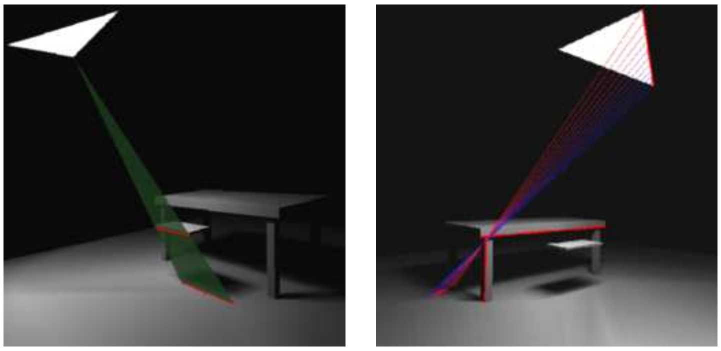
Prenons par exemple la scène figure 1. Le calcul du maillage de discontinuité de cette scène prend environ deux minutes, ce qui exclut l'interactivité si nous voulons déplacer un objet. Nous proposons deux méthodes de mise à jour dynamique du maillage de discontinuité lorsqu'un objet se déplace de façon discrète dans une scène composée de polygones. Nous appellerons *objet dynamique* l'objet se déplaçant et *objets statiques* les autres objets de la scène. L'éclairage initial de la scène est calculé par l'algorithme décrit par [5]. La première méthode s'appuie sur la cohérence spatiale autour du mouvement d'un objet, l'autre sur la cohérence de visibilité. Ces deux méthodes permettent de réutiliser le maillage de discontinuité et l'éclairage calculé initialement pour une grande partie de la scène, et de modifier uniquement la région affectée par le mouvement de l'objet. En considérant le déplacement de la tablette (qui se trouve sous le bureau) dans une scène contenant 200 polygones (voir figure 1(a) et (b)), nous arrivons avec les nouvelles méthodes à mettre à jour l'éclairage direct en temps approchant l'interactivité.

Après une brève introduction sur les maillages de discontinuité, la section suivante présente les travaux précédents réalisés sur la mise à jour de l'éclairage. Dans la section 3, nous décrivons la première méthode basée sur la cohérence spatiale et dans la section 4 celle basée sur la cohérence de visibilité. Les résultats sont présentés dans la section 5.

2 État de l'art

2.1 Maillage de discontinuité

Le maillage de discontinuité est construit à partir d'intersections de surfaces planes ou quadriques avec les objets de la scène. La figure 1(c) illustre un tel maillage. Ces surfaces décrivent les changements de visibilité et sont appelées des *surfaces de discontinuité*. Elles sont définies par l'interaction des arêtes et des sommets de la scène. On distingue les surfaces de discontinuité planes EV construites à partir d'une arête et d'un sommet, des surfaces quadriques EEE composées de trois arêtes (exemple figure 2). D'autres types de surfaces existent et correspondent à des cas dégénérés. Les *surfaces de discontinuité maximales* sont les surfaces de discontinuité délimitant la zone d'ombre de la région éclairée. Elles entourent toutes les surfaces de discontinuité liées au même objet. À chaque face du maillage ainsi construit est associée une structure de données appelée *projection inverse* dans laquelle est stockée la topologie de la source visible depuis chaque face. Cette structure permet de déterminer la valeur de l'éclairage de chacune des faces de façon très efficace car aucun calcul de visibilité n'est nécessaire.



(a) (b)

FIG. 2 - *(a) Exemple de surface EV (b) Exemple de surface EEE*

2.2 Méthodes antérieures de mise à jour des environnements dynamiques

La méthode de radiosité [14] permet d'obtenir un rendu réaliste de l'éclairage global d'une scène complexe. Son application directe aux environnements dynamiques est limitée aux changements de point de vue. Plusieurs méthodes ont été développées afin de pouvoir l'employer pour des environnements dynamiques. La méthode de Baum [2] consiste à précalculer l'éclairage de la scène sans l'objet dynamique. La mise à jour de la scène lors du mouvement de l'objet consiste à localiser les changements de visibilité puis à modifier les valeurs de l'éclairage affectées par le mouvement. George [7] et Chen [3] ont présenté plus tard des approches basées sur la radiosité progressive pour accélérer l'affichage du résultat. Enfin Shaw [13] a proposé une technique utilisant la radiosité hiérarchique pour permettre une mise à jour efficace grâce aux informations de visibilité correspondant aux liens hiérarchiques. Le point commun de ces méthodes est la localisation spatiale des changements induits par le mouvement d'un objet. Leur principal inconvénient provient du fait que la mise à jour de l'éclairage reste approximative et de nombreux calculs sont inutiles.

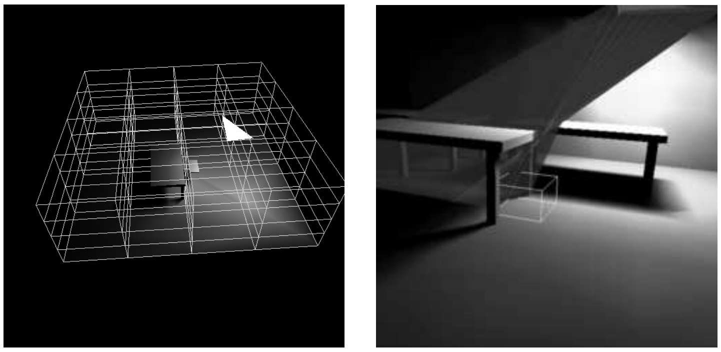
La mise à jour des ombres a été abordée par Chrysanthou et al. [4] qui utilisent des arbres BSP mais ne considèrent que les sources ponctuelles. Worall et al. [16] ont esquissé les modifications d'un maillage de discontinuité seulement pour des sources ponctuelles, et ils s'intéressent plus à la mise à jour de la triangulation dont le coût est pourtant négligeable par rapport à celui du maillage de discontinuité. Aucun travail complet n'a donc été jusqu'ici réalisé sur la mise à jour du maillage de discontinuité pour les sources étendues lors du déplacement d'un objet.

3 Utilisation de la cohérence spatiale pour la mise à jour du maillage de discontinuité

3.1 Stockage des surfaces de discontinuité dans la subdivision spatiale

La scène est décomposée en une grille régulière (voir figure 3(a)) [1]. Cette grille est utilisée dans l'algorithme initial de construction du maillage afin de localiser rapidement les polygones de la scène pour la propagation des surfaces de discontinuité [5], en évitant ainsi l'intersection des surfaces de discontinuité avec la plupart des polygones de la scène.

Dans notre nouvelle méthode, lors du calcul initial de l'éclairage, nous stockons en plus dans chaque voxel la liste des surfaces de discontinuité qui le traversent. La figure 3(b) illustre un voxel affiché avec les surfaces de discontinuité correspondantes.



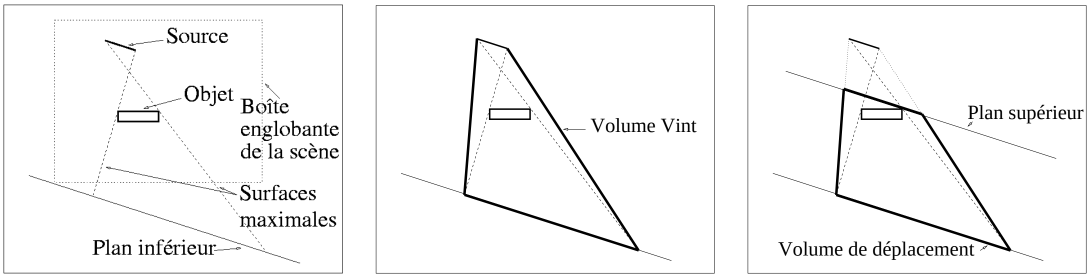
(a) (b)

FIG. 3 - *(a) Grille régulière (b) Voxel avec les surfaces de discontinuité qui le traversent*

3.2 Localisation par un volume de déplacement

Lors du déplacement d'un objet dynamique, l'ombre de l'objet va changer, ainsi qu'éventuellement la projection des ombres des objets statiques qui interagissent avec l'objet dynamique. Les modifications à faire se trouvent dans la région délimitée par l'objet lui-même et son ombre, donc en terme d'événements par les surfaces de discontinuité maximales de l'objet dynamique. Pour localiser les modifications, un *volume de déplacement* est construit (exemple figure 6(a)). Sa construction, illustrée figure 4, se fait en trois étapes. Premièrement la boîte englobante de la scène sert à définir un plan inférieur parallèle à la source. Un polygone de ce plan est délimité par la projection des surfaces maximales liées à l'objet dynamique (voir figure 4(a)). Ensuite un volume intermédiaire appelé Vint est construit à partir des deux polygones correspondant au plan inférieur et à la source (voir figure 4(b)). Le volume de déplacement est enfin obtenu en coupant le volume Vint par un plan parallèle à la source et contenant le point de l'objet dynamique le plus proche de la source (voir figure 4(c)). Ce volume entoure donc les surfaces de discontinuité maximales et est délimité par deux plans parallèles à la source : le plan inférieur à la scène et celui supérieur à l'objet dynamique.

Grâce au volume de déplacement ainsi défini, nous déterminons les structures affectées par le mouvement qui sont les surfaces de discontinuité liées à l'objet dynamique, le maillage de discontinuité, les projections inverses et les valeurs de l'éclairage. Dans l'algorithme de mise à jour, le volume de déplacement entoure les deux positions de l'objet dynamique.



(a) (b) (c)

FIG. 4 - *(a) Construction du plan inférieur (b) Premier volume Vint (c) Volume de déplacement final*

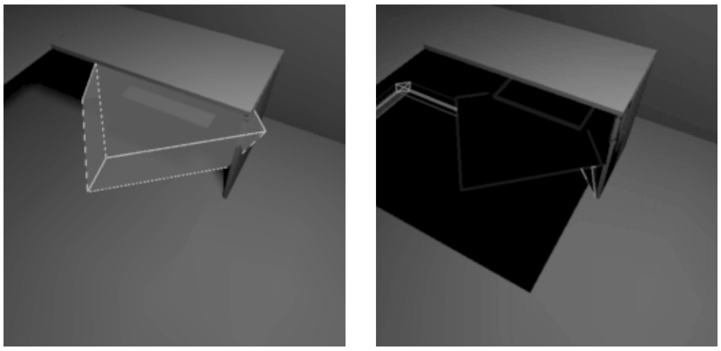
3.3 Algorithme de mise à jour

L'algorithme 5 décrit les actions effectuées pour la mise à jour de l'éclairage lors du déplacement d'un objet. On commence par déplacer l'objet dynamique puis on construit le volume de déplacement autour de ses deux positions.

Traitement dynamique (Objet Obj_{dyn})
début
déplacerObjet (Obj_{dyn}, de la position pos1 à pos2)
construireVolume
localiserInformationsDansGrille
majMaillage
majBPIncrementale
majEclairage
fin

FIG. 5 - *Procédure générale des actions de mise à jour dynamique lors du déplacement d'un objet*

Le volume de déplacement permet de localiser les parties des polygones qui lui sont intérieures sur lesquelles le maillage est effacé en coupant si nécessaire les arêtes partiellement incluses (voir figure 6 (b)). Ensuite, les surfaces de discontinuité statiques détectées dans les voxels contenus ou intersectant le volume de déplacement sont repropagées. Puis les surfaces de discontinuité dynamiques sont reconstruites et repropagées en utilisant la liste de polygones contenus dans le volume Vint. Pour chaque arête insérée dans le maillage, les faces qui l'entourent sont marquées « modifiées ». Les projections inverses de chacune des faces modifiées sont ensuite recalculées avec l'algorithme incrémental développé pour la méthode initiale de calcul de l'éclairage [5, 15]. Il est important de noter que le calcul des projections inverses est local et très rapide. Enfin nous mettons



(a) (b)

FIG. 6 - *(a) Volume de déplacement autour de la tablette (b) Polygones sur lesquels le maillage est effacé*

à jour les valeurs de l'éclairage direct de chacune des faces modifiées. De plus amples détails sont donnés dans [12].

	Scène 1			Scène 2			Scène 3		
Position	t_{DDS}	t_{BPI}	t_{TD}	t_{DDS}	t_{BPI}	t_{TD}	t_{DDS}	t_{BPI}	t_{TD}
Pos1	1.63	0.28	1.95	1.72	0.28	2.04	6.47	0.52	7.05
Pos2	2.75	0.74	3.53	2.80	0.81	3.65	5.70	0.68	6.44
Pos3	3.51	0.48	4.04	3.97	0.51	4.53	10.51	0.39	10.96
Pos4	2.74	0.48	3.26	3.39	0.52	3.95	10.59	0.42	11.07

TAB. 1 - *Répartition du temps d'exécution de la première méthode*

En comparaison avec l'algorithme qui consisterait à recalculer l'éclairage de la scène entière à chaque pas de temps, nous avons limité le nombre de surfaces statiques à repropager, le temps nécessaire à la création des surfaces dynamiques en évitant la recherche des polygones, ainsi que le nombre de faces du maillage effectivement modifiées à l'intérieur du volume de déplacement.

La table 1 illustre la répartition du temps en secondes passé lors de l'exécution de cette méthode sur les trois scènes de test décrites section 5. Le temps t_{TD} est le temps total de mise à jour. Le temps t_{BPI} passé dans le calcul des projections inverses est bien moins important que celui de la propagation des surfaces de discontinuité t_{DDS}. En tenant compte de ce résultat, la deuxième méthode présentée dans la section suivante permet de diminuer considérablement ce temps de propagation.

4 Méthode incrémentale de mise à jour du maillage basée sur l'analyse des changements de visibilité

Les résultats de la méthode précédente montrent que la plupart du temps de calcul est passé dans la propagation des surfaces (table 1). Cette propagation se fait en plusieurs étapes. Nous avons remarqué que le calcul des intersections des surfaces avec les polygones de la scène était le plus coûteux.

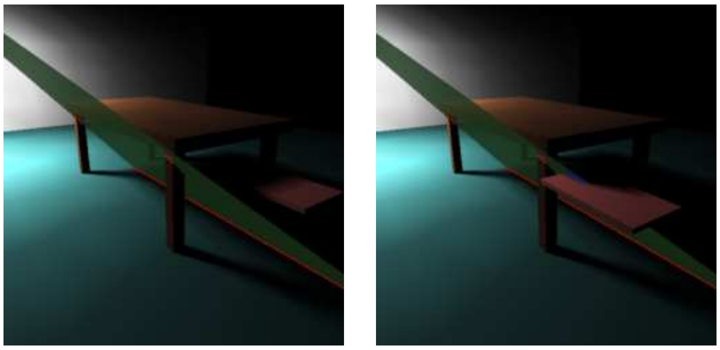
Un nouvel algorithme incrémental basé sur la cohérence de visibilité remplace la procédure *majMaillage* de l'algorithme précédent (voir figure 5) par la procédure *majMaillageIncrémentale* décrite figure 7. Afin de limiter les calculs, la *liste des intersections* avec les polygones de la scène est stockée avec chacune des surfaces de discontinuité . Ensuite les arêtes du maillage et les surfaces de discontinuité sont traitées différemment selon si elles sont *statiques* (liées aux objets statiques) (section 4.1) ou *dynamiques* (liées à l'objet dynamique) (section 4.2). Les détails des algorithmes de ces méthodes sont donnés dans [11].

majMaillageIncrémentale
définition des variables
$Poly2D$: Liste de polygones 2D
$Poly$: Liste des polygones contenu dans le volume
$V_{englobant}$: Volume de déplacement
$aretes_{dyn}$, $aretes_{stat}$: Liste d'arêtes dynamiques et statiques à traiter
$Interact$: Booléen
DS_{stat} : Liste des surfaces statiques trouvées dans les voxels
DS_{dyn} : Liste des surfaces dynamiques
début
  $Poly2D \leftarrow$ *intersecte* ($V_{englobant}$, $Poly$)
  pour chaque polygone 2D P de $Poly2D$ **faire**
    pour chaque arête A de P **faire**
      si estDynamique(A) **alors**
        insérer ($aretes_{dyn}$,A)
      sinon
        $Interact \leftarrow$ *traiterArêteStatique* (A)
        si $Interact$ **alors** *insérer* ($aretes_{stat}$, A)
    traiterArêteDynamique ($aretes_{dyn}$)
    effacerArêteStatique($aretes_{stat}$)
    effacerArêteDynamique ($aretes_{dyn}$)
  traiterSurfacesStatiques (DS_{stat})
  traiterMaillageDynamique (DS_{dyn}, Obj_{dyn}, $Poly2D$)
fin

FIG. 7 - *Mise à jour incrémentale du maillage*

4.1 Traitement des surfaces de discontinuité statiques

Les seuls changements intervenant pour les arêtes statiques du maillage proviennent des modifications des intersections de la surface de discontinuité statique correspondante avec l'objet dynamique avant ou après son déplacement (voir figure 8(a)(b)).



(a) (b)

FIG. 8 - *(a) Intersection d'une surface de discontinuité avec les polygones de la scène (b) Intersection de la même surface après le mouvement de la tablette*

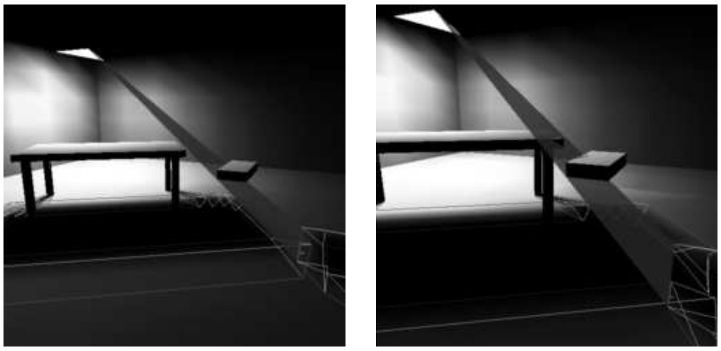
Si une arête du maillage est trouvée sur un polygone intérieur au volume de déplacement et si la surface de discontinuité correspondante n'intersecte pas l'objet dynamique, cette arête du maillage n'est pas effacée et la surface de discontinuité correspondante n'est pas repropagée. Le nombre de surfaces statiques traitées est ainsi diminué. Par contre, chaque arête provenant d'une surface de discontinuité qui intersecte l'objet dynamique est effacée du maillage. La liste des intersections de la surface de discontinuité correspondante est mise à jour : seules les intersections avec l'objet dynamique sont recalculées. Pour chacune des surfaces en interaction avec l'objet dynamique, la visibilité est ensuite réévaluée et les arêtes correspondantes insérées dans le maillage. Par rapport à la méthode précédente, il n'y a plus de propagation totale des surfaces statiques. Le coût important de la recherche des polygones et le calcul des intersections est ainsi évité. Seule une intersection avec l'objet dynamique est nécessaire.

4.2 Traitement des surfaces dynamiques

Le traitement des surfaces de discontinuité dynamiques se limite pour l'instant aux surfaces de discontinuité arête-sommet (EV) et à la construction-destruction des surfaces de discontinuité triple arêtes (EEE) dont une arête de la source. Cette partie est en cours d'implémentation.

4.2.1 Traitement des surfaces dynamiques EV

Le traitement des surfaces dynamiques EV se fait par analyse du maillage sur chacun des polygones intérieurs au volume de déplacement. À chaque arête EV dynamique est associé un polygone 2D délimitant les positions initiale et finale de cette arête. Dans cette région, toutes les arêtes statiques EV sont parcourues afin de détecter si elles sont liées soit au même sommet V soit à la même arête E d'un polygone. Si une telle arête statique est trouvée, cela signifie qu'un changement de visibilité a eu lieu. Une modification de l'intersection de la surface dynamique avec le polygone auquel appartient cette arête statique doit être réalisée.



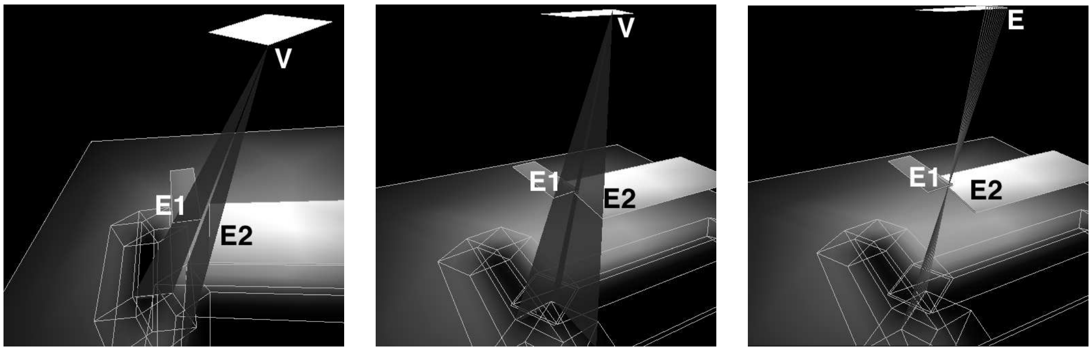
(a) (b)

FIG. 9 - *(a) Surface EV dynamique (b) Changement de visibilité et modification des intersections de la surface EV après le mouvement de la tablette*

Par exemple figure 9(a)(b), lorsque l'objet se déplace, la surface EV va intersecter le dessus de la table. Ce changement de visibilité peut se détecter grâce aux surfaces EV construites avec une arête de la table et le même sommet de la source que la surface dynamique. Une nouvelle intersection avec le polygone de la table est calculée et insérée dans la liste des intersections stockées avec la surface dynamique.

4.2.2 Construction et destruction des surfaces dynamiques EEE

En analysant les modifications du maillage dynamique, il est possible de déterminer à quel moment une surface EEE est créée et quand elle doit être détruite. On utilise pour cela les détections de changements de visibilité décrites dans la section précédente. Soit une arête du maillage liée à une surface E_1V telle que l'arête E_1 appartient à l'objet dynamique et V à la source. Si lors du déplacement elle en vient à intersecter une arête statique du maillage E_2V où V est le même sommet de la source, alors deux surfaces EE_1E_2 et $E'E_1E_2$ sont crées où E et E' sont les deux arêtes de la source adjacentes à V. La figure 10 illustre la création des deux surfaces triple arêtes après le déplacement d'une tablette détectées par l'apparition d'une intersection de deux arêtes EV. Réciproquement, lorsque deux arêtes EV composées avec le même sommet de la source ne s'intersectent plus, les arêtes EEE correspondantes sont détruites.

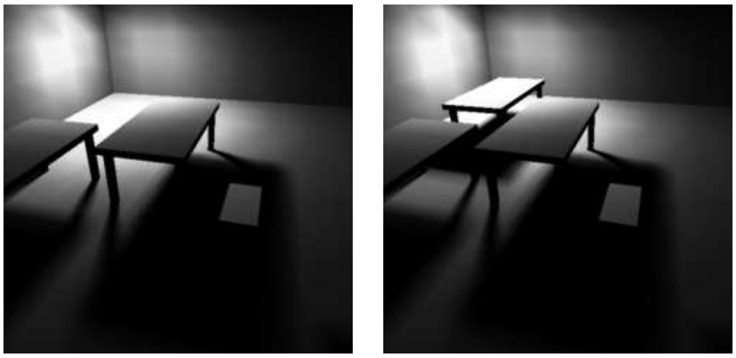


(a) (b) (c)

FIG. 10 - *(a) Deux surfaces EV qui ne s'intersectent pas (b) Intersection des surfaces EV et (c) Une des deux surfaces triple arêtes créées*

5 Résultats

Les deux algorithmes ont été testés sur trois scènes différentes dans lesquelles on déplace par interaction au clavier une tablette. La scène 1 contient deux tables (voir figure 11(a)), la scène 2 trois tables (voir figure 11(b)), et la scène 3 un bureau avec deux tiroirs et une chaise (voir figure 1). Elles sont composées respectivement de 72, 108, et 200 polygones. Pour chacune des scènes, la tablette se déplace en 4 positions successives. Les tests ont été réalisés sur une machine Indigo 2 (R4400, 150Mhz).



(a) (b)

FIG. 11 - *Les deux premières scènes de test: (a) Scène 1, (b) Scène 2*

Dans la table 2 nous présentons les résultats pour les trois scènes. Le coût mémoire ajouté pour les méthodes proposées est représenté en Ko par m_{DS} pour le stockage des surfaces de discontinuité dans la grille et m_{IL} pour le stockage de la liste des intersections des surfaces de discontinuité (uniquement réalisé pour la deuxième méthode). Ce coût de stockage augmente en fonction de la complexité des scènes mais reste raisonnable pour ces trois scènes. Le pourcentage n_{DS} de surfaces de discontinuité utilisées par rapport au nombre total de surfaces de discontinuité dans les algorithmes de mise à jour est respectivement en moyenne de 16%, 10% et 6% pour les scènes 1,2 et 3; celui n_f du nombre de faces modifiées de 12%, 10% et 4%. Il n'y a donc qu'une région très restreinte pour laquelle les calculs sont refaits.

Les temps t_{TS}, t_{TD1} et t_{TD2} sont respectivement les temps en secondes du calcul complet du maillage à chaque pas de temps, de la première méthode dynamique et de la deuxième méthode dynamique. s_1 et s_2 sont les facteurs d'accélération des deux méthodes par rapport au temps t_{DS} nécessaire pour le calcul total de la scène. Les temps de calcul pour la première méthode sont

Position	n_{DS}	n_f	m_{DS}	m_{IL}	t_{TS}	t_{TD1}	s_1	t_{TD2}	s_2
Pos1	9	14.5	19.57	4.50	15.04	1.95	7.71	1.59	9.4
Pos2	15	14	19.68	5.19	15.64	3.53	4.43	2.48	6.3
Pos3	21	12	19.75	5.75	15.53	4.04	3.83	2.32	6.7
Pos4	21	11	19.75	6.31	15.20	3.26	4.60	1.45	10.4

Résultats pour la scène 1

Position	n_{DS}	n_f	m_{DS}	m_{IL}	t_{TS}	t_{TD1}	s_1	t_{TD2}	s_2
Pos1	5	11	37.34	6.81	27.27	2.04	13.30	1.79	15.2
Pos2	8	10	37.44	7.50	28.38	3.65	7.74	2.77	10.2
Pos3	13	9	37.51	8.06	27.84	4.53	6.14	2.54	10.9
Pos4	14	9	37.51	8.62	27.76	3.95	7.03	1.63	17.0

Résultats pour la scène 2

Position	n_{DS}	n_f	m_{DS}	m_{IL}	t_{TS}	t_{TD1}	s_1	t_{TD2}	s_2
Pos1	4	4.5	97.65	20.25	114.70	7.05	16.26	5.15	22.27
Pos2	6	4.7	97.65	20.25	113.99	6.44	17.70	5.62	20.38
Pos3	7	4.7	99.53	20.25	116.25	10.96	10.96	5.70	20.39
Pos4	7	4.7	99.54	20.25	117.01	11.07	10.57	5.60	20.89

Résultats pour la scène 3

TAB. 2 - *Résultats*

suffisants pour permettre l'interaction dans les scènes 1 et 2. Par contre, les temps trop importants de la scène 3 confirment la nécessité de la deuxième méthode. La deuxième méthode permet une interaction relative pour une scène complexe telle que la scène 3. Les temps de calculs sont rapides, 1.8 à 2.7 secondes pour une mise à jour à chaque pas de temps de la scène 2 ce qui rend possible une interaction avec l'utilisateur. Pour une scène plus complexe comme la scène 3, nous gagnons un facteur 22 d'accélération par rapport au calcul complet du maillage à chaque pas de temps.

Les résultats sont prometteurs bien que l'implémentation courante soit loin d'être optimisée. Avec l'implémentation de la mise à jour incrémentale des surfaces dynamiques, nous espérons arriver à une mise à jour de 1 seconde pour les scènes 1 et 2, et obtenir un temps où l'interaction est très acceptable pour la scène 3.

6 Perspectives et Conclusion

Les méthodes présentées permettent la mise à jour interactive du maillage de discontinuité et de l'éclairage direct d'une scène composée de polygones et d'une source étendue. Elles peuvent cependant être encore améliorées.

Le coût mémoire peut être limité de manière significative avec l'utilisation d'une grille adaptative. Ensuite, pour un pas de déplacement variable, un critère de choix de l'utilisation alternative soit d'un volume de déplacement englobant les deux positions, soit de deux volumes correspondant à chacune d'elles, peut être développé afin d'adapter la taille de la région à mettre à jour en fonction de la taille du déplacement. La deuxième méthode ouvre une voie vers une méthode complète pour détecter exactement les seuls changements nécessaires, en examinant exclusivement le maillage 2D sur les polygones. Cette dernière méthode serait réalisée pour un déplacement continu et un environnement fermé. De plus, une prédiction du temps de mise à jour serait possible en déterminant un critère de complexité. Il devrait être fonction de toute information ayant rapport avec le maillage comme par exemple le nombre de surfaces de discontinuité se trouvant dans la région du déplacement de l'objet dynamique, ou le nombre de polygones. L'importance de chacune des surfaces de discontinuité par rapport à la précision de l'ombre pourrait être mesurée [8]. Un résultat approximatif dépendant de la justesse désirée serait obtenu plus rapidement. Enfin, les méthodes présentées pourraient être appliquées dans un contexte de maillage pour les éclairages globaux [6].

Remerciements

Nous tenons à remercier Seth Teller pour sa suggestion de traitement des surfaces statiques, ainsi que Frédo Durand pour ses idées et ses discussions sur les approches des surfaces dynamiques.

Merci aussi à Claude Puech pour son soutien, et à Nicolas Tsingos d'avoir aidé à la réalisation des images.

Références

[1] John Amanatides and Andrew Woo. A fast voxel traversal algorithm for ray tracing. In *Eurographics '87*, pages 3–10. North-Holland, August 1987.

[2] Daniel R. Baum, John R. Wallace, Michael F. Cohen, and Donald P. Greenberg. The back-buffer algorithm : an extension of the radiosity method to dynamic environments. *The Visual Computer*, 2:298–306, 1986.

[3] Shenchang Eric Chen. Incremental radiosity: An extension of progressive radiosity to an interactive image synthesis system. *Computer Graphics*, 24(4):135–144, August 1990. Proceedings SIGGRAPH '90 in Dallas (USA).

[4] Yiorgos Chrysanthou and Mel Slater. Shadow volume BSP trees for computation of shadows in dynamic scenes. In Pat Hanrahan and Jim Winget, editors, *1995 Symposium on Interactive 3D Graphics*, pages 45–50. ACM SIGGRAPH, April 1995. ISBN 0-89791-736-7.

[5] George Drettakis and Eugene Fiume. A fast shadow algorithm for area light sources using back projection. In *Computer Graphics Proceedings, Annual Conference Series:* SIGGRAPH '94 (Orlando, FL), pages 223–230. ACM SIGGRAPH, New York, July 1994.

[6] George Drettakis and Francois Sillion. Accurate visibility and meshing calculations for hierarchical radiosity. In *Proceedings of 7th Eurographics Workshop on Rendering in Porto, Portugal (Rendering Techniques '96, Pueyo, Shroeder eds., Springer Verlag)*. Eurographics, June 1996.

[7] David W. George, François Sillion, and Donald P. Greenberg. Radiosity redistribution for dynamic environments. *IEEE Computer Graphics and Applications*, 10(4), July 1990.

[8] Stephen Hardt and Seth Teller. High-fidelity radiosity rendering at interactive rates. In *Proceedings of 7th Eurographics Workshop on Rendering in Porto, Portugal (Rendering Techniques '96, Pueyo, Shroeder eds., Springer Verlag)*. Eurographics, June 1996.

[9] Paul Heckbert. Discontinuity meshing for radiosity. *Third Eurographics Workshop on Rendering*, pages 203–226, May 1992.

[10] Dani Lischinski, Filippo Tampieri, and Donald P. Greenberg. Discontinuity meshing for accurate radiosity. *IEEE Computer Graphics and Applications*, 12(6):25–39, November 1992.

[11] Céline Loscos. Cohérence de visibilité pour les maillage de discontinuité des scènes dynamiques. Master's thesis, UJF, Grenoble, France, septembre 1996. Mémoire de magistère 3eme année.

[12] Céline Loscos. Cohérence spatiale pour les maillage de discontinuité des scènes dynamiques. Master's thesis, INPG, Grenoble, France, juin 1996. Mémoire de DEA.

[13] Erin Shaw. Hierarchical radiosity for dynamic environments. Master's thesis, Cornell University, Ithaca, New York, August 1994.

[14] François Sillion and Claude Puech. *Radiosity and Global Illumination*. Morgan Kaufmann publishers, San Francisco, 1994.

[15] A. James Stewart and Sherif Ghali. Fast computation of shadow boundaries using spatial coherence and backprojections. In Andrew Glassner, editor, *Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, Computer Graphics Proceedings, Annual Conference Series, pages 231–238. ACM SIGGRAPH, ACM Press, July 1994. ISBN 0-89791-667-0.

[16] Adam Worrall, Claire Willis, and Derek Paddon. Dynamic discontinuities for radiosity. In Harold P. Santo, editor, *Edugraphics + Compugraphics Proceedings*, pages 367–375, Alvor, Portugal '95, December 12 1995. GRASP- Graphic Science Promotions & Publications, P.O. Box 4076, Massama, 2745 Queluz, Portugal.