

INTRODUCTION

En 1968, Arthur Appel publia un article sur la synthèse d'images qui portait en germe la technique du lancer de rayon que Turner Whitted popularisa en 1980. Depuis, vingt huit années fertiles en idées et travaux ont considérablement enrichi l'informatique graphique aussi bien dans la recherche d'un plus grand réalisme que dans l'amélioration des performances des algorithmes à tel point que les images de synthèse sont parvenues au stade industriel et sont très largement utilisées aussi bien pour le cinéma, la conception de pièces mécaniques, l'architecture, la médecine, les simulateurs (aviation, sport, ...), la visualisation scientifique (météorologie, géologie, dynamique des fluides, ...), etc... De plus, l'amélioration prévisible de la qualité des images par des effets de rendu encore plus réalistes et la chute des temps de calculs permettent d'entrevoir des applications totalement nouvelles par l'introduction de l'animation en temps réel et l'interaction avec un monde partiellement ou totalement numérique.

Cependant, on constate aujourd'hui une diminution des travaux de recherche concernant les optimisations de calcul du rendu réaliste en général et du lancer de rayon en particulier. De plus, une grande partie de ces publications concerne des champs d'application très particuliers ou des études de rendu réaliste sur des architectures particulières : machines dédiées à la synthèse d'images, ordinateurs parallèles ou encore distribution sur des réseaux de stations de travail. Il est donc légitime de se demander si les recherches de méthodes d'accélération stagnent parce que tout a été dit, que ce domaine a désormais atteint son stade de maturité et que

seules des améliorations du matériel permettront de calculer des images de grande qualité en temps réel, ou s'il reste des pistes originales à explorer.

Parallèlement à ces travaux d'optimisation des programmes de synthèse d'images, se pose le très important problème de l'évaluation de la qualité de ces logiciels, tant au niveau des performances que de la qualité de l'image produite. Quelques essais à base de "benchmarks" ont été tentés, cependant il n'existe pas à ce jour de protocole de tests général permettant de faire une bonne évaluation des différentes approches et au-delà, il n'existe pas de modèle purement mathématique d'une méthode d'optimisation du lancer de rayon. Or sans un tel protocole, il n'est pas possible de répondre à la question de savoir si la recherche algorithmique de solution pour le lancer de rayon est un domaine clos ou non.

Ce mémoire est consacré à l'optimisation de l'algorithme du lancer de rayon et à la recherche d'un protocole d'évaluation des performances de cet algorithme. Les développements qui y sont liés reposent sur l'hypothèse d'être le moins dépendants possible des technologies logicielles et matérielles actuelles.

Les chapitres I et II présentent les modèles physiques sur lesquels se base le lancer de rayon.

Nous exposons un état de l'art des nombreuses méthodes d'accélération tant séquentielles que parallèles dans la première moitié du chapitre III. Un ensemble de définitions qui permettent de construire une classification nouvelle et importante des méthodes d'accélération du lancer de rayon est présenté dans la seconde partie de ce chapitre. Cette nouvelle classification est basée sur la décomposition du problème du lancer de rayons en trois graphes décrivant la structure de la scène (graphe des objets réels), la connaissance liée aux arbres de calcul des pixels (graphe des objets virtuels) et enfin la structure algorithmique du programme (graphe des processus). À la différence de la classification classique des méthodes d'optimisation du lancer de rayon, notre approche inclut naturellement le parallélisme comme approche originale d'optimisation. Elle permet également de décrire qualitativement une approche particulière d'optimisation.

Ce chapitre théorique introduit en outre toutes les définitions qui seront nécessaires pour décrire notre méthode d'optimisation, et ceci en dehors de toute contrainte matérielle.

À partir de ces définitions, un mécanisme de déduction et un ensemble de règles originales définissent la méthode d'accélération parallèle qui est le cœur de cette thèse (chapitre IV). Ces règles ont pour but de reconstruire une connaissance topologique globale de la scène à partir des calculs sur les pixels effectués localement par aux processus. L'écran étant décomposé en zones de pixels associées à un processus et les calculs se faisant dans un ordre très strict, il est en effet possible de reconstruire cette connaissance en fonction des objets vus ou non vus localement par chaque processus ainsi que les objets vus ou non vus par les autres processus. Les processus échangent donc des messages sur les objets de la scène qu'ils ont découverts, les envoient dans une direction incluant un ensemble de processus (Nord Est par exemple) et appliquent des règles de déductions en fonction des informations échangées, ce qui génère de nouveaux messages permettant de ne plus tester de nombreux objets de la scène lors des intersections rayons/objets. Ce mécanisme de déduction et de coopération est un résultat important de nos travaux.

Les chapitres V, VI et VII proposent et exploitent un formalisme qui permet de valider notre méthode et au-delà, de montrer l'intérêt voire la nécessité d'une analyse précise des performances en infographie.

Au chapitre V, nous faisons un rapide état de l'art des approches classiques d'analyse de performances et nous analysons le lancer de rayon de façon à extraire les paramètres qui sont importants et significatifs de la justesse et de la performance d'un programme de lancer de rayon.

Au chapitre VI, nous présentons un protocole de test qui compare deux lancers de rayon et qui essaie au maximum de ne pas à avoir recours à des indices de performances liés à la machine (comme le temps de calcul), ce qui nous permet de réaliser une analyse de performance qui tente d'aller un peu plus loin que les classiques "benchmarks".

Le chapitre VII montre qu'il est envisageable d'aller au-delà de la simple analyse basée sur un protocole de tests sur machine comme celui que nous présentons aux chapitres V et VI ; une méthode mathématique a été développée pour analyser la complexité réelle d'un lancer de rayon de premier niveau intégrant une partie des optimisations présentées dans cette thèse. Une telle démarche, donnant certaines mesures de la complexité de l'algorithme, nous semble importante car elle permet d'évaluer des performances sans avoir recours à l'implantation machine. C'est un des éléments importants de notre approche. Cette approche statistique débouche sur des applications nouvelles des statistiques d'ordre et du calcul des variations. Ces méthodes classiques en probabilité et en mécanique trouvent là un champ nouveau d'application. En retour, ces domaines bien connus

des mathématiques nous permettent d'analyser finement notre algorithme. En effet, il est par exemple possible d'établir des formules mathématiques permettant de prévoir le gain (en nombre d'objets non testés par rayon) apporté par un ensemble donné d'optimisations (i.e. une version simplifiée de notre lancer de rayon). On peut également étudier ces formules générales de façon à rechercher, pour une scène donnée, le meilleur découpage en processus de l'écran par exemple. Les résultats obtenus par cette démarche de modélisation mathématique sont importants et novateurs. Cependant beaucoup de travaux devront encore être faits pour améliorer et compléter cette approche (en particulier pour inclure les rayons secondaires et d'ombrage).

Des compléments sur l'environnement logiciel que nous avons créé (modeleur, calcul de texture, lancer de rayon, environnement de parallélisation - chapitre VII) et un album de nos images sont également donnés (chapitre X).

Enfin, nous terminerons au chapitre IX sur une étude rapide de notre approche au cas du rendu réaliste en temps réel. Dans ce cas, les règles d'optimisation sont toujours applicables et peuvent même être un peu améliorées.

La conclusion de ce mémoire concernera l'apport des travaux de recherche développés dans cette thèse et sur les perspectives qu'ils ouvrent.

CHAPITRE I

Présentation générale du Lancer de Rayon

Mots clés

Lancer de rayon, image de synthèse.

Résumé

Nous présentons en quelques pages les principes informatiques liés à la méthode dite du lancer de rayon.

Summary

We present in a few pages the general principles of ray-tracing.

I) Introduction

L'infographie a eu très tôt l'ambition de produire des images réalistes, c'est-à-dire, dans l'absolu, des images synthétiques que l'on ne puisse pas différencier des photographies. Cette marche vers la réalité fut la plupart du temps réalisée par touches successives en se demandant ce qu'il manquait aux résultats qu'il était possible d'obtenir avec l'état de l'art d'une époque. Ainsi de la fruste représentation en fil de fer des origines, les chercheurs ont produit des algorithmes de parties cachées style Z-buffer ou Scan-line, puis améliorèrent ces algorithmes pour pouvoir générer des ombres ou tenir compte des effets lumineux de diffusion, de texture ou encore des réflexions et des transmissions pour des objets comme les miroirs ou les vitres. Les méthodes les plus qualitativement satisfaisantes aujourd'hui sont la radiativité d'une part, et le lancer de rayon d'autre part. C'est sur cette dernière que nous allons nous pencher.

II) Principe du lancer de rayon

Au début des années 60, les informaticiens s'avisèrent que les modèles proposés en optique géométrique par les mathématiciens et les physiciens étaient d'excellents candidats pour la génération d'images de synthèse [Ap68]. Hélas, la technologie de l'époque ne permit pas de dépasser le stade théorique de ce problème et il fallut attendre la fin des années 70, sinon la décennie suivante pour être enfin capable de calculer des images que l'on puisse décentement qualifier de "réalistes".

III) Le modèle de l'appareil photo simplifié

À la base de l'idée du lancer de rayon, il y a un modèle d'appareil photo simplifié (Cf. Figure I.1). Imaginons une boîte totalement fermée. Sur un côté de celle-ci, on plaque une feuille photosensible qui servira de pellicule et sur le côté opposé, on perce un minuscule trou (sténopé) avec une aiguille, de façon à ce que chaque point de l'espace à l'extérieur de la boîte ne puisse envoyer qu'un photon à travers ce trou. Le fait que le trou soit très petit évite d'avoir une pellicule surexposée car chaque point de celle-ci sera en rapport avec un unique point de l'espace (symbolisé sur notre dessin par un segment). On remarquera que l'image est inversée par rapport à ce qui est vu [Gl89].

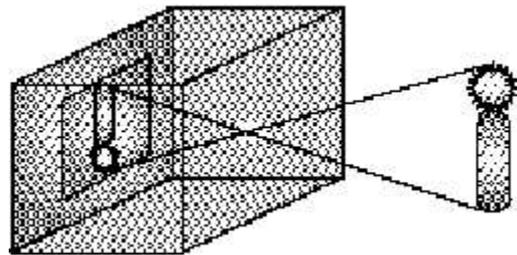


Figure I-1

IV) L'algorithme

Les informaticiens utilisent ce modèle en image de synthèse, en plaçant à l'endroit du trou d'aiguille un point spécial (l'œil de l'observateur) et en interposant un écran "transparent" entre cet œil et l'image (l'écran de l'ordinateur). Le principe de l'algorithme devient alors très simple (Cf. Figure I.2).

L'écran étant décomposé en pixels, on va calculer un vecteur directeur entre chacun d'eux et l'œil. Ce vecteur va donner l'équation d'une droite qui sera appelée "rayon" car elle simule le trajet d'un rayon qui irait d'un point quelconque de l'espace vers l'œil, tout en passant par le pixel choisi. Cependant, à la différence de la réalité, ce rayon va de l'œil vers la scène et non le contraire.

Une fois que l'on a obtenu le vecteur directeur d'un rayon, on calcule si celui-ci intercepte un objet de la scène (par un calcul élémentaire de géométrie).

S'il n'intercepte rien, alors le rayon se perd dans l'espace et la couleur du pixel sera celle de la lumière ambiante. Sinon, on calcule la couleur de l'objet intercepté au point d'interception. Celle-ci tient compte des effets d'ombrage (on relance un rayon

en direction des sources lumineuses pour savoir si l'on est dans une ombre) et de diffusion. Ces paramètres peuvent être déterminés localement. Si l'objet n'est ni réfléchissant ni transparent, la couleur du pixel est la couleur calculée localement.

Si l'objet est transparent ou réfléchissant, alors on calcule avec les lois de Descartes (Cf. chapitre sur les modèles d'illumination) le vecteur réfléchi et/ou transmis. A ce moment on obtient un ou plusieurs nouveaux rayons et l'on peut recommencer récursivement le même calcul. Celui-ci se termine lorsqu'un rayon se perd à l'infini ou lorsque l'on touche un objet non spéculaire ou encore lorsque l'on atteint la limite de récursivité.

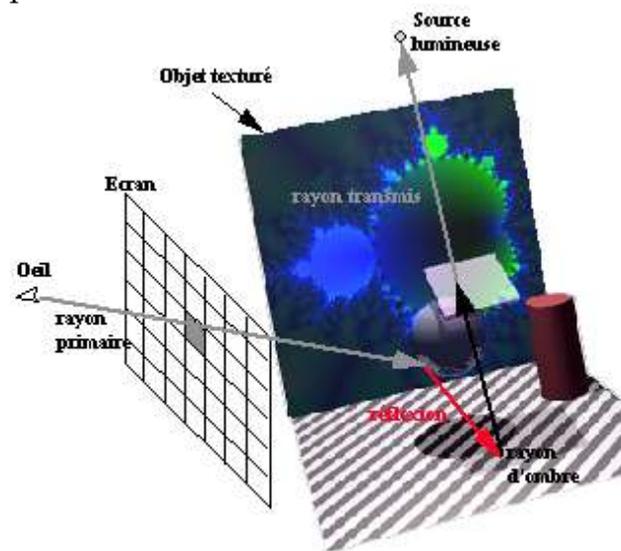


Figure I-2

```

Fonction calcul_couleur(rayon)    couleur ;
Début

    couleur_locale = couleur_ambiante ;
    couleur_r = couleur_t = (0,0,0) ;
    Pour chaque objet faire
        Calculer_intersection(rayon,objet) ;
        Garder_l'objet_le_plus_proche ;
    Fait ;

    Si un objet a été gardé alors
        Calculer couleur_locale ;

        Si réflexion Alors
            couleur_r=calcul_couleur(rayon_réfléchi) ;
        Fsi

        Si transmission Alors
            couleur_t=calcul_couleur(rayon_transmis) ;
        Fsi

    Fsi

    retourner(couleur_locale+couleur_r+couleur_t);

Fin

Programme principal
début

    Pour chaque pixel faire
        rayon=Calculer_vecteur(œil, pixel) ;
        afficher(calcul_couleur(rayon) ) ;
    Fait

Fin.

```

Bibliographie

- [Ap68] : APPEL (A.). - "Some techniques for shading machine renderings of solids."
- AFIPS, Spring Joint Computer conf. 1968, p. 32-37.
- [Gl89] : An introduction to ray tracing/ sous la direction de Andrew S. Glassner. -
Londres : Academic press limited, 1989. 327 p.