Chapitre 2

Balayage de graphe de visibilité

Dans ce chapitre nous considérons le graphe et le complexe de visibilité de scènes polygonales. Nous exposons d'abord un algorithme optimal de calcul de graphe de visibilité et étudions ensuite ses performances expérimentales. Nous montrons ensuite comment utiliser cet algorithme, qui effectue en fait un balayage du complexe de visibilité, pour construire le complexe.

— *Dis Astérix, nous ne sommes tout de même pas venus ici pour balayer leur pays !*

Astérix et les Goths.

2.1 Calcul optimal de graphe de visibilité

Nous étudions dans cette section un algorithme optimal de calcul de graphe de visibilité. Nous formalisons d'abord le problème de balayage à l'aide de la notion de *filtre*. Nous introduisons ensuite *les arbres d'horizon*, outil qui permet d'effectuer ce balayage. Nous étudions enfin la complexité théorique de l'algorithme.

2.1.1 Balayage topologique

Comme nous l'avons indiqué dans le chapitre précédent, nous considérons des scènes composées de polygones simples, ayant deux à deux une intersection nulle. Nous considérons aussi que les polygones sont des objets composites et que les objets élémentaires considérés dans les relations de visibilité (appelés simplement « objets ») sont en fait les segments de droites qui forment les côtés des polygones. Pour rester cohérent dans les relations de visibilité, nous considérons qu'un rayon partant à l'infini voit en fait l'objet O_∞, qui peut être considéré comme le « ciel bleu » qui entoure la scène.

Nous notons $(\mathcal{V}, \mathcal{E})$ Le graphe de visibilité de la scène, n son nombre de sommets ($n = |\mathcal{V}|$) et m son nombre d'arêtes ($m = |\mathcal{E}|$). Nous étendons l'ensemble des arêtes géométriques du graphe de visibilité à l'ensemble des arêtes orientées u du graphe de visibilité : $u = ((p,q), \theta)$ est l'arête orientée de p à q de pente θ où $\theta \equiv \text{angle}(\overrightarrow{p,q})(2\pi)$. Cet ensemble – maintenant infini – est aussi noté $\mathcal{E}$ (le contexte permettra de distinguer l'ensemble des arêtes géométriques de l'ensemble des arêtes orientées).

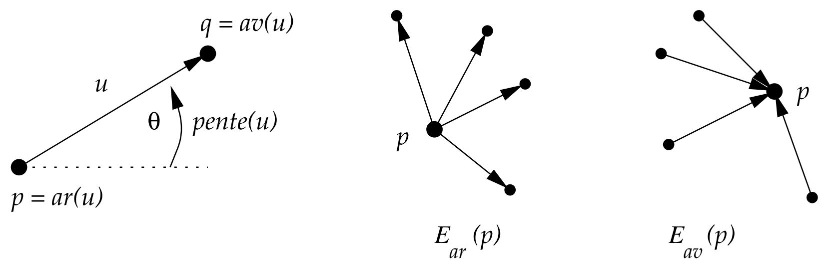

FIG. 2.1 – *Notations pour les arêtes.*

Pour une arête $u = ((p,q), \theta)$, nous notons (figure 2.1) :

- $ar(u) = p$, le sommet arrière de u ;
- $av(u) = q$, le sommet avant de u ;
- $pente(u) = \theta$, la pente de u.

Pour un point p de la scène nous notons aussi :

- $\mathcal{E}_{ar}(p) = \{u \in \mathcal{E} / ar(u) = p\}$, l'ensemble des arêtes partant de p ;
- $\mathcal{E}_{av}(p) = \{u \in \mathcal{E} / av(u) = p\}$, l'ensemble des arêtes arrivant en p ;
- $\mathcal{E}(p) = \mathcal{E}_{ar}(p) \cup \mathcal{E}_{av}(p)$, l'ensemble des arêtes ayant p pour extrémité.

Nous avons vu dans le chapitre précédent qu'Edelsbrunner et Guibas [EG89] et Overmars et Welzl [OW88] définissaient le balayage rotationnel du graphe de visibilité comme devant, pour chaque sommet du graphe, reporter les arêtes incidentes dans l'ordre de leur pente.

Nous nous sommes inspiré de la formalisation de Pocchiola et Vegter [PV96a] pour présenter ici la notion de balayage topologique. Nous introduisons d'abord un ordre sur l'ensemble $\mathcal{E}$ des arêtes du graphe.

Définition 2.1 *Soit un sommet p de la scène. La relation $\prec_p$ définie par*

$$\forall (u,v) \in \mathcal{E}(p)^2,\ u \prec_p v \iff \mathit{pente}(u) < \mathit{pente}(v),$$

est un ordre total sur $\mathcal{E}(p)$.

La relation $\prec$ définie par

$$\forall (u,v) \in \mathcal{E},\ u \prec v \iff \exists\, (u_i)_{0 \le i \le k},\ u_0 = u,\ u_k = v,\ u_{i-1} \text{ et } u_i \text{ ont une extrémité commune } p \text{ et } u_{i-1} \prec_p u_i,$$

est un ordre partiel sur $\mathcal{E}$.

Une arête u est donc inférieure à une arête v si on peut passer de u à v par une suite d'arêtes incidentes et de pente croissante. Il est facile de vérifier que cette relation est bien une relation d'ordre partiel sur $\mathcal{E}$, compatible avec l'ordre total induit par l'ordre sur les pentes des arêtes. Nous étendons cet ordre aux ensembles d'arêtes : pour deux sous-ensembles U et V d'arêtes de $\mathcal{E}$, $U \prec V$ signifie que pour toute paire d'arêtes $(u,v) \in U \times V$, soit $u \prec v$, soit u et v ne sont pas comparables.
Deux arêtes incidentes sont donc comparables, mais aussi

Propriété 2.2 *Deux arêtes sécantes sont comparables suivant l'ordre partiel $\prec$.*

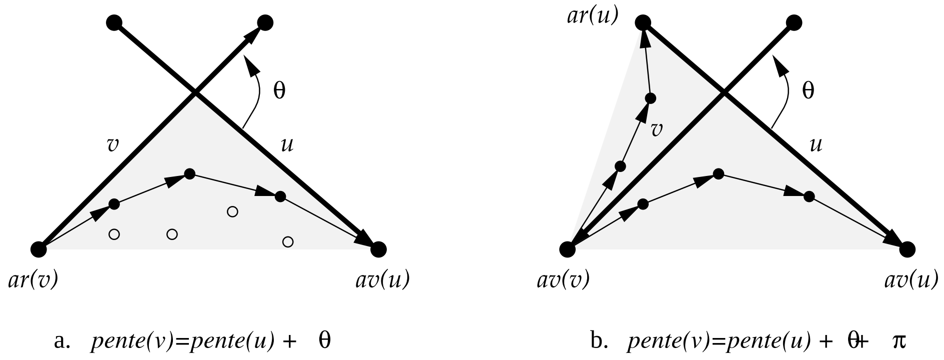

FIG. 2.2 – *Comparaison de deux arêtes sécantes.*

Démonstration. Soient deux arêtes sécantes u et v. Supposons que $\mathit{pente}(v)$ soit supérieure à $\mathit{pente}(u)$ mais inférieure à $\mathit{pente}(u) + \pi$ (figure 2.2a). Considérons les arêtes joignant les sommets de l'enveloppe convexe des points contenus dans le triangle formé par $av(u)$, $ar(v)$ et le point d'intersection des deux arêtes. Alors on peut passer de u à v en suivant ces arêtes, qui constituent une suite d'arêtes de pentes croissantes.

Si $\mathit{pente}(v)$ est supérieure à $\mathit{pente}(u) + \pi$ (figure 2.2b), alors nous passons de u à $v-\pi$ comme précédemment, puis de $v-\pi$ à $u+\pi$ par l'enveloppe convexe du triangle $(av(v), v \cap u, ar(u))$, puis de $u+\pi$ à v par la première enveloppe convexe. □

Faire un balayage topologique du graphe de visibilité revient à faire un tri topologique des arêtes du graphe selon l'ordre $\prec$ qui vient d'être défini. Pour exprimer ce tri nous utilisons la notion de *filtre* :

un filtre R d'un ensemble partiellement ordonné E est un sous-ensemble de E tel que $\forall x \in R$, $x \prec y \Rightarrow y \in R$.

Nous définissons alors le balayage topologique en terme de filtre :

Définition 2.3 *Un balayage topologique du graphe de visibilité $(\mathcal{V}, \mathcal{E})$ est une suite de filtres $(R_i)_{0 \leq i \leq k}$ de $\mathcal{E}$ tels que $\forall\ 0 \leq i < k$, $R_{i+1} = R_i \backslash \{u\}$ où $u \in \min\limits_{\prec} R_i$.*

Le filtre R_i représente l'ensemble des arêtes qui restent à balayer alors que les arêtes de $\mathcal{E} \setminus R_i$ ont déjà été vues. Nous pouvons initialiser le balayage en prenant par exemple pour R_0 l'ensemble des arêtes de pente supérieure à un angle θ_0.
Pour faire progresser le balayage nous devons trouver une arête minimale u parmi les arêtes restantes du filtre, et nous disons alors que nous *passons* l'arête u.

Nous pouvons alors calculer le graphe de visibilité de la scène en balayant l'ensemble de ses arêtes géométriques, c'est-à-dire en balayant ses arêtes orientées de pente comprise entre θ_0 et $\theta_0 + \pi$.

2.1.2 Arbres d'horizon

Dans cette partie nous montrons comment calculer $\min_\prec R_i$, l'ensemble des arêtes minimales de R_i. À cet effet nous introduisons deux nouveaux graphes : l'*arbre d'horizon* inférieur (resp. supérieur). L'étude de leurs propriétés permet de donner un algorithme pour calculer $\min_\prec R_i$.

Étant donné que pour tout point p la relation $\prec_p$ est un ordre total sur $\mathcal{E}(p)$, nous pouvons désigner l'élément suivant et l'élément précédent selon cet ordre. Pour une arête u de $\mathcal{E}(p)$, nous notons alors

- $succ^s_p(u) = \min\limits_{\prec_p}\{v \in \mathcal{E}_{ar}(p)/u \prec_p v\}$: arête suivante de u dans $\mathcal{E}_{ar}(p)$;
- $pred^s_p(u) = \max\limits_{\prec_p}\{v \in \mathcal{E}_{ar}(p)/v \prec_p u\}$: arête précédente de u dans $\mathcal{E}_{ar}(p)$;
- $succ^i_p(u) = \min\limits_{\prec_p}\{v \in \mathcal{E}_{av}(p)/u \prec_p v\}$: arête suivante de u dans $\mathcal{E}_{av}(p)$;
- $pred^i_p(u) = \max\limits_{\prec_p}\{v \in \mathcal{E}_{av}(p)/v \prec_p u\}$: arête précédente de u dans $\mathcal{E}_{av}(p)$.

Une arête peut être le successeur de plusieurs arêtes ; en effet soit u une arête de $\mathcal{E}_{ar}(p)$, alors pour toute arête v de $\mathcal{E}(p)$ telle que $u \prec_p v \prec_p succ^s_p(u)$, on a $succ^s_p(v) = succ^s_p(u)$. En revanche, si l'on se restreint à $\mathcal{E}_{ar}(p)$, l'ensemble des arêtes comprises entre une arête u et son successeur $succ^s_p(u)$ est vide.

Étant donné un filtre R, les éléments de $\min_\prec R$ se trouvent parmi les successeurs des éléments de $\mathcal{E} \backslash R$. Dans cette optique nous définissons deux nouveaux graphes, les *arbres d'horizon* :

Définition 2.4 *Soit un filtre R de $\mathcal{E}$ et soit $B = \mathcal{E} \backslash R$. Le graphe $\mathcal{H}^s(R) = (\mathcal{V}^s, \mathcal{E}^s)$ défini par*

– $\mathcal{V}^s = succ^s(B) \setminus B = \{u \in R \;/\; \exists\, v \in B, u = succ^s_{ar(v)}\}$,
– $\mathcal{E}^s = \{(u, v) \in \mathcal{V}^{s2} / av(u) = ar(v) \text{ et } pente(u) < pente(v) + \pi\}$,

est un arbre de racine $\max\limits_{\prec} \mathcal{V}^s$, appelé arbre d'horizon supérieur. Le parent de u dans cet arbre est noté $\mathcal{E}^s(u)$.
Similairement, l'arbre d'horizon inférieur est l'arbre défini par

– $\mathcal{V}^i = succ^i(B) \setminus B = \{u \in R \;/\; \exists\, v \in B, u = succ^i_{av(v)}\}$,
– $\mathcal{E}^i = \{(u, v) \in \mathcal{V}^{i2} / ar(u) = av(v) \text{ et } pente(u) < pente(v) + \pi\}$,

et le parent de u dans cet arbre est noté $\mathcal{E}^i(u)$.

Démonstration. S'il existe un cycle d'arêtes consécutives de $\mathcal{V}^s$, alors il existe deux arêtes consécutives u et v de ce cycle telles que $pente(u) \geq pente(v) + \pi$, et donc ce cycle n'est pas un cycle dans le graphe $\mathcal{H}^s$. □

La figure 2.3 montre les arbres d'horizon d'une scène où R est l'ensemble des arêtes du graphe de visibilité de pente supérieure à $-\pi/2$.

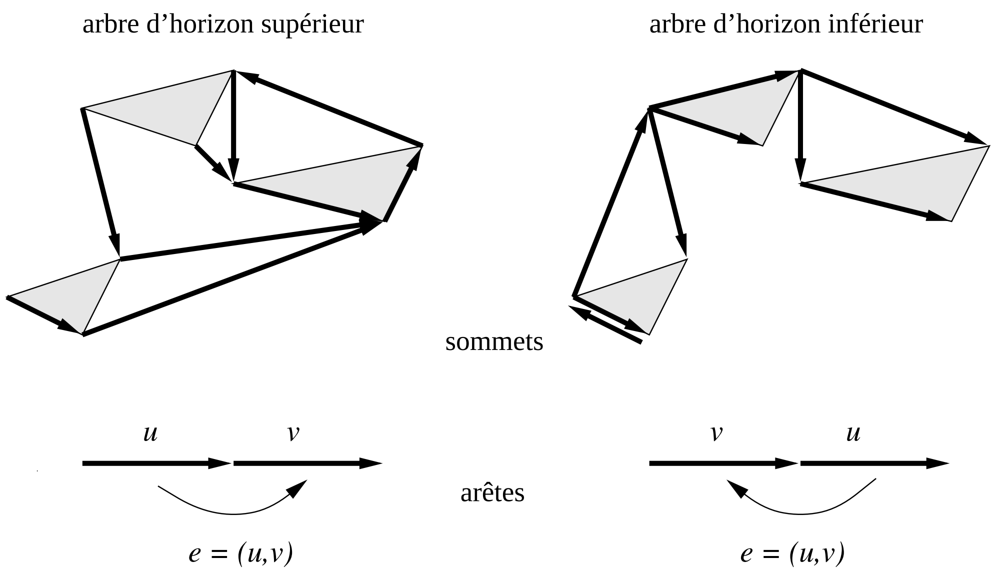

FIG. 2.3 – *Arbres d'horizon.*

Dans la suite de ce chapitre, une « arête de l'arbre d'horizon » désignera en fait – sauf mention contraire explicite – une arête du graphe de visibilité appartenant à l'arbre d'horizon, c'est-à-dire une arête de $\mathcal{E}$ appartenant à $\mathcal{V}^s$ (resp. $\mathcal{V}^i$).

Deux arêtes arrivant à un même point ont le même parent et sont dites sœurs. Elles sont classées par pente croissante, ce qui permet de désigner pour une arête sa sœur gauche (de pente plus petite) et sa sœur droite (de pente plus grande).

Les deux arbres d'horizon sont liés par une relation de symétrie : si nous inversons le sens des arêtes de l'arbre d'horizon inférieur associé à R, alors nous obtenons l'arbre d'horizon supérieur associé à $R' = R+\pi$, l'ensemble des arêtes de R auxquelles on a appliqué une rotation d'angle π. Dans la suite, nous ne démontrerons certaines propriétés que pour les arbres d'horizon supérieurs : les démonstrations sont à une symétrie près les mêmes pour les arbres d'horizon inférieurs.

Les arbres d'horizon permettent de caractériser les arêtes déjà balayées :

Propriété 2.5 *Soit une arête $u = (p, q)$ du graphe de visibilité, alors $\forall v \in \mathcal{E}_{ar}(p)$*

- *si $u \in R$, alors $u \prec_p v \Rightarrow v \notin \mathcal{V}^s$;*
- *si $u \in \mathcal{V}^s$, alors $v \prec_p u \iff v \notin R$.*

Démonstration. Si l'arête $u \in R$ est inférieure à l'arête $v \in \mathcal{E}_{ar}(p)$, alors $succ^s(u) \preceq v$, ce qui empêche l'existence d'une arête $w \notin R$ (et donc inférieure à u) telle que $v = succ^s(w)$. et donc v n'appartient pas à l'arbre d'horizon.

Soit $u \in \mathcal{V}^s$. D'après le point précédent on a $v \in R \Rightarrow u \preceq_p v$, soit $v \prec_p u \Rightarrow v \notin R$. La réciproque est une conséquence directe de la propriété des filtres. □

Ces propriétés combinatoires des arêtes des arbres d'horizon induisent dans la scène certaines propriétés géométriques :

Propriété 2.6 *Toutes les arêtes d'un arbre d'horizon ont deux à deux une intersection nulle ou dégénérée (extrémité commune ou support commun).*

Démonstration. Soient deux arêtes u et v du graphe de visibilité ayant une intersection non nulle et non dégénérée. Reprenons la première situation de la démonstration de la propriété 2.2 (figure 2.2a). Soit w l'arête l'enveloppe convexe issue de $ar(v)$. Si u appartient à l'arbre d'horizon, alors comme w est supérieure à u, w appartient au filtre R et donc d'après la propriété 2.5, v n'appartient pas à l'arbre d'horizon. La démonstration est similaire pour la deuxième situation. □

Certaines propriétés sont plus liées aux relations de visibilité entre objets :

Propriété 2.7 *Soient deux arêtes u et v consécutives appartenant à l'arbre d'horizon supérieur ($(u, v) \in \mathcal{V}^{s2}$, $av(u) = ar(v)$), alors*

- *$v \prec u \Rightarrow (ar(u), av(v)) \notin \mathcal{E}$; (figure 2.4a)*
- *$u \prec v \Rightarrow pred^s_{av(u)}(v) \prec_{av(u)} u$;*
- *Si $u \prec v$ et $av(u) \neq ar(u)^+$, alors les rayons issus de $av(u)$ et de directions respectives u et v sont stoppés par un même segment. De plus le triangle délimité par ces deux rayons et ce segment est vide de tout sommet de la scène.(figure 2.4b)*

Démonstration. Supposons que $v \prec u$ et notons $w = (ar(u), av(v))$. Comme w est supérieure à v, w appartient à R (propriété des filtres). Si w est une arête du graphe de visibilité alors, comme $w \prec_{ar(u)} u$, d'après la propriété 2.5 (2[e] point) w n'appartient pas à R ce qui est absurde, w n'est donc pas une arête du graphe de visibilité.

Supposons que $u \prec v$ et notons $w = pred^s_{av(u)}(v)$. Comme $v \in \mathcal{V}^s$, d'après la

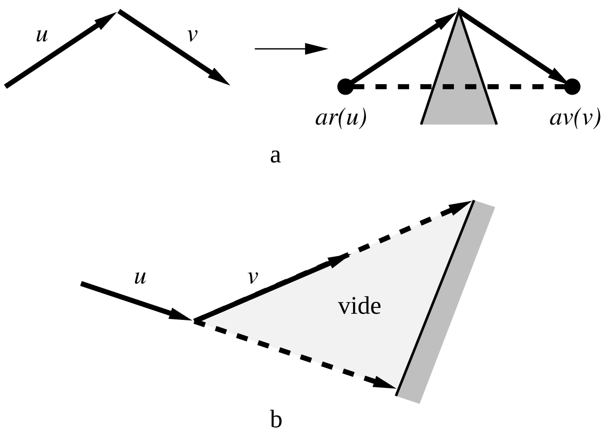

FIG. 2.4 – *Propriétés de visibilité des arbres d'horizon.*

propriété 2.5, w n'appartient pas à R. Si w était supérieure à u, alors w appartiendrait à R (propriété des filtres), ce qui est contradictoire, donc w est inférieure à u.

Enfin le dernier point est une conséquence directe du deuxième point. □

Nous remarquons que nous pouvons associer à chaque sommet de la scène l'unique arête de l'arbre d'horizon supérieur (resp. inférieur) partant de (resp. arrivant à) ce sommet :

Proposition 2.8 *Soit un filtre R et soit $B = \mathcal{E} \backslash R$. L'application e^s (resp. e^i) définie par les deux définitions équivalentes*

- $e^s(p) = \min_{\prec_p}(R \cap \mathcal{E}_{ar}(p))$ *(resp.* $e^i(p) = \min_{\prec_p}(R \cap \mathcal{E}_{av}(p))$*)*
- $e^s(p) = succ^s_p(\max_{\prec_p}(B \cap \mathcal{E}(p)))$ *(resp.* $e^i(p) = succ^i_p(\max_{\prec_p}(B \cap \mathcal{E}(p)))$*)*

est une bijection de $\mathcal{V}$ sur $\mathcal{V}^s$ (resp. $\mathcal{V}^i$).

Démonstration. Comme $e^s(p) \in \mathcal{E}_{ar}(p)$, e^s est injectif.

Soit $u \in \mathcal{V}^s$, et $p = ar(u)$. La propriété 2.5 permet d'affirmer que $\forall w \in \mathcal{E}_{ar}(p)$, $w \in R \Rightarrow u \preceq_p w$, ce qui signifie que $u = \min_{\prec_p}(R \cap \mathcal{E}_{ar}(p))$; u a donc $ar(u)$ pour antécédent par e^s : e^s est surjectif. □

Grâce à cette bijection nous connaissons la taille des arbres d'horizon : $|\mathcal{V}^s| = n$ et $|\mathcal{E}^s| = n$.

Nous pouvons aussi réexprimer la relation de parenté : le parent de u dans $\mathcal{H}^s$ est $\mathcal{E}^s(u) = e^s(av(u))$, et le parent de u dans $\mathcal{H}^i$ est $\mathcal{E}^i(u) = e^i(ar(u))$.

Nous énonçons maintenant le théorème qui montre comment les arbres d'horizon permettent de déterminer l'ensemble des arêtes pouvant être passées pour faire progresser le balayage :

Théorème 2.9 *Soit un filtre R, alors*

$$\min_{\prec} R = \{u \in \mathcal{V}^s \cap \mathcal{V}^i / u \prec \mathcal{E}^s(u) \text{ et } u \prec \mathcal{E}^i(u)\}.$$

Démonstration. Soit une arête $u = (p, q)$ appartenant à l'ensemble qui vient d'être défini. Comme u appartient à $\mathcal{V}^s$, alors $u = \min_{\prec_p}(R \cap \mathcal{E}_{ar}(p))$. D'autre part, $u \prec \mathcal{E}^i(u)$ se réécrit $u \prec e^i(p) \prec_p \min_{\prec_p}(R \cap \mathcal{E}_{av}(p))$, ce qui entraîne $u = \min_{\prec_p}(R \cap \mathcal{E}(p))$. Similairement nous montrons que $u = \min_{\prec_q}(R \cap \mathcal{E}(q))$, et donc finalement u appartient à $\min_\prec R$.

L'inclusion inverse est évidente. □

Nous pouvons effectuer le calcul de $\min_\prec R$ directement en étudiant chaque arête d'un des deux arbres d'horizon. Le temps de calcul est alors linéaire ($O(n)$). Cependant si nous utilisons cette méthode lors du balayage des arêtes du graphe de visibilité, le calcul complet de $\min_\prec R$ à chaque passage coûterait un temps total $O(mn)$, ce qui est trop coûteux pour calculer le graphe de visibilité. Pour abaisser ce coût, nous ne calculons directement $\min_\prec R$ que pour initialiser le balayage et nous utilisons ensuite le théorème seulement pour mettre à jour $\min_\prec R$ lors de chaque passage.

2.1.3 Mise à jour de $\min\limits_{\prec} R$ lors d'un passage

Nous supposons que nous passons du filtre R_t au filtre R_{t+1} en passant l'arête $u = (p, q)$ du graphe de visibilité. Nous supposons aussi que les arbres d'horizon ont été mis à jour et nous notons u^s (resp. u^i) l'arête qui remplace u dans l'arbre d'horizon supérieur (resp. inférieur) : $\mathcal{V}^s_{t+1} = (\mathcal{V}^s_t \setminus \{u\}) \cup \{u^s\}$ (resp. $\mathcal{V}^i_{t+1} = (\mathcal{V}^i_t \setminus \{u\}) \cup \{u^i\}$). Pour mettre à jour $\min_\prec R$, nous devons chercher les nouvelles arêtes à rajouter à $\min_\prec R_t \setminus \{u\}$.

Nous devons d'abord considérer les arêtes des nouveaux arbres d'horizon qui n'étaient pas présentes dans les anciens arbres, c'est-à-dire les nouvelles arêtes u^s et u^i. D'après l'égalité $\min_\prec R = \{u \in \mathcal{V}^s \cap \mathcal{V}^i / u \prec \mathcal{E}^s(u) \text{ et } u \prec \mathcal{E}^i(u)\}$, nous devons aussi considérer les arêtes des arbres d'horizon dont le père a changé, c'est-à-dire les arêtes qui avaient u pour père. Parmi celles-ci il suffit de ne considérer que les deux arêtes de pente minimale $w^s = \min_{\prec_p} {\mathcal{E}^s_{t+1}}^{-1}(p)$ et $w^i = \min_{\prec_q} {\mathcal{E}^i_{t+1}}^{-1}(q)$.

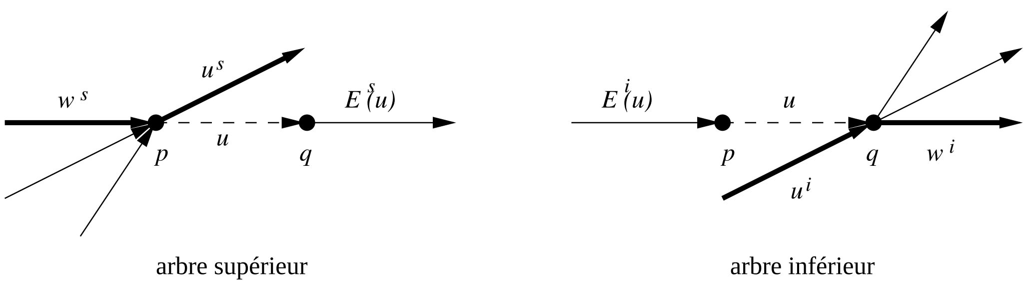

FIG. 2.5 – *Correspondance entre les mises à jour dans les arbres d'horizon.*

Si nous dessinons les deux arbres d'horizon au voisinage de ces arêtes (figure 2.5), nous voyons qu'il pourrait exister des correspondances entre les arêtes des deux arbres d'horizon, notamment entre w^s et $\mathcal{E}^i_t(u)$ (resp. entre w^i et $\mathcal{E}^s_t(u)$). En effet, $\mathcal{E}^i(u)$ est inchangée et donc $\mathcal{E}^i_t(u) = \min_{\prec_p}\{v \in R_{t+1} \cap \mathcal{E}_{av}(p)\}$; d'autre part, l'arête de pente minimale w^s vérifie $w^s = \min_{\prec_p}\{v \in \mathcal{V}^s_{t+1} \cap \mathcal{E}_{av}(p)\}$, ce qui entraîne $\mathcal{E}^i_t(u) \preceq w^s$. Si

$\mathcal{E}_t^i(u)$ appartient à $\mathcal{V}_{t+1}^s$, d'après la propriété 2.5 nous avons $w^s \preceq \mathcal{E}_t^i(u)$ et alors w^s et $\mathcal{E}_t^i(u)$ désignent la même arête. Le même raisonnement s'applique à w^i et $\mathcal{E}_t^s(u)$.

Au lieu de calculer w^s (resp. w^i), nous considérons directement $\mathcal{E}_t^i(u)$ (resp. $\mathcal{E}_t^s(u)$), et l'algorithme de mise à jour s'écrit :

$\min\limits_{\prec} R = \min\limits_{\prec} R \setminus \{u\}$
si $\mathcal{E}_t^i(u) \prec u^s$
  si $\mathcal{E}_t^i(u) \in \mathcal{V}_{t+1}^s$ **et** $\mathcal{E}_t^i(u) \prec \mathcal{E}_{t+1}^i(\mathcal{E}_t^i(u))$
    $\min\limits_{\prec} R = \min\limits_{\prec} R \cup \{\mathcal{E}_t^i(u)\}$
sinon
  si $u^s \in \mathcal{V}_{t+1}^i$ **et** $u^s \prec \mathcal{E}_{t+1}^s(u^s)$
    $\min\limits_{\prec} R = \min\limits_{\prec} R \cup \{u^s\}$
si $\mathcal{E}_t^s(u) \prec u^i$
  si $\mathcal{E}_t^s(u) \in \mathcal{V}_{t+1}^i$ **et** $\mathcal{E}_t^s(u) \prec \mathcal{E}_{t+1}^s(\mathcal{E}_t^s(u))$
    $\min\limits_{\prec} R = \min\limits_{\prec} R \cup \{\mathcal{E}_t^s(u)\}$
sinon
  si $u^i \in \mathcal{V}_{t+1}^s$ **et** $u^i \prec \mathcal{E}_{t+1}^i(u^i)$
    $\min\limits_{\prec} R = \min\limits_{\prec} R \cup \{u^i\}$

Nous devons montrer maintenant comment calculer u^s et u^i, c'est-à-dire comment mettre à jour les arbres d'horizon.

2.1.4 Mise à jour des arbres d'horizon

Nous étudions la mise à jour de l'arbre d'horizon supérieur lors du passage de l'arête $u = (p, q)$ (la mise à jour de l'arbre inférieur est similaire à une symétrie près).

Cette mise à jour s'effectue en considérant successivement les quatre cas décrits ci-après. Les deux premiers cas se traitent directement, les deux autres cas demandent de considérer une nouvelle structure : les *chaînes d'arêtes*.

Premier cas : $av(u) = ar(u)^+$

L'arête rentre dans le polygone, et l'arête suivante est l'arête $u^s = (ar(u), ar(u)^-)$ sortant du polygone (figure 2.6a).

Deuxième cas : $av(\mathcal{E}^s(u)) = ar(u)$

Nous en déduisons que le demi-plan supérieur délimité par la droite passant par u est vide. u est alors une arête de l'enveloppe convexe de la scène, et la prochaine arête u^s est l'arête de l'enveloppe convexe de la scène partant de $ar(u)$ (figure 2.6b).

Troisième cas : $av(u)^-$ est au-dessus de u

Considérons l'enveloppe convexe inférieure des points contenus dans le triangle $(ar(u), av(u), av(u)^-)$ ($av(u)$ exclus). La prochaine arête est l'arête de cette enveloppe

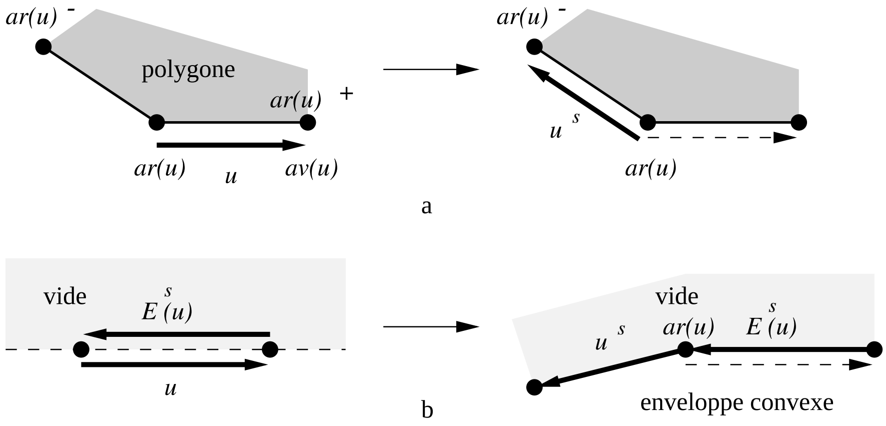

FIG. 2.6 – *Cas directs de mise à jour des arbres d'horizon.* **a.** *Rentrée dans un polygone.* **b.** *Sortie hors de la scène.*

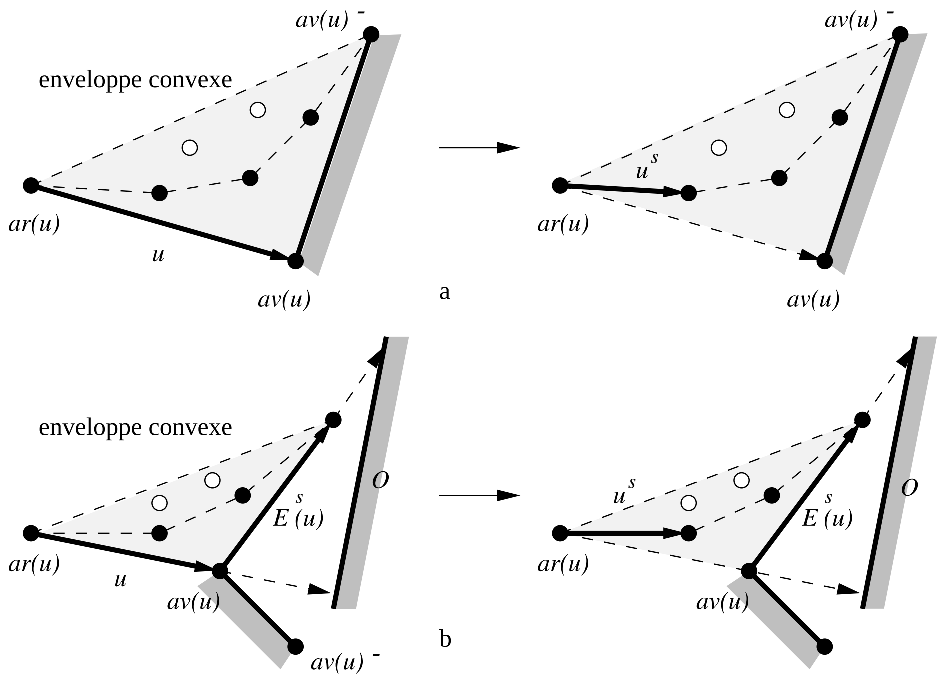

FIG. 2.7 – *Mise à jour des arbres d'horizon.* **a.** *Rayon passant par u stoppé par le segment incident à $av(u)$.* **b.** *Rayon passant par u stoppé après $av(u)$.*

partant de $ar(u)$ (figure 2.7a).

Quatrième cas : $av(u)^{-}$ est en-dessous de u

Soit O le segment de la scène vu depuis $av(u)$ dans la direction u. Le triangle, délimité par O et les deux rayons issus de $av(u)$, de directions respectives u et $\mathcal{E}^s(u)$

et stoppés par O, est vide (propriété 2.5).

Considérons l'enveloppe convexe inférieure des points contenus dans le triangle $(ar(u), av(u), av(\mathcal{E}^s(u)))$ ($av(u)$ exclus). La prochaine arête est l'arête de cette enveloppe partant de $ar(u)$ (figure 2.7b).

Nous avons une situation similaire dans les deux derniers cas : nous devons considérer une enveloppe convexe de points. Mais cette enveloppe convexe est en fait déjà calculée :

Proposition 2.10 *Toutes les arêtes de l'enveloppe convexe considérée dans le troisième ou le quatrième cas – exceptée celle partant de p – appartiennent à l'arbre d'horizon.*

Démonstration. Considérons un point r de l'enveloppe convexe (différent des deux points extrêmes), et notons $cv(r)$ l'arête de l'enveloppe convexe partant de r. Comme $cv(r)$ est supérieure à u, elle appartient à R et donc l'arête $e^s(r)$ de l'arbre d'horizon issue de r vérifie $e^s(r) \preceq cv(r)$.
Si $e^s(r)$ avait une pente inférieure à celle de $cv(r)$, d'après la géométrie de la situation, la pente de $e^s(r)$ serait inférieure à celle de u, ce qui est impossible ($u \in \min_{\prec} R$) ; les deux arêtes $e^s(r)$ et $cv(r)$ sont donc les mêmes. □

L'enveloppe convexe considérée fait partie de plus grands sous-ensembles d'arêtes de l'arbre d'horizon :

Définition 2.11 *On appelle chaîne d'arêtes d'un arbre d'horizon $\mathcal{H}^s$, toute suite maximale d'arêtes $(u_i)_{0 \leq i \leq k}$ telle que*

- $\forall\, 0 \leq i < k \quad u_i \in \mathcal{V}^s$ *et* $av(u_i) = ar(u_{i+1})$ *;*
- $\forall\, 0 \leq i < k \quad av(u_i)^-$ *est au-dessus de* u_i *;*
- $\forall\, 0 < i \leq k \quad ar(u_i)^+$ *est au-dessus de* u_i *;*
- $\forall\, 0 \leq i < k \quad u_i$ *est l'arête de plus petite pente de* $\mathcal{V}^s$ *arrivant en* $av(u_i)$*.*

De cette définition nous en déduisons les propriétés géométriques suivantes :

Propriété 2.12 *Soit une chaîne d'arêtes $(u_i)_{0 \leq i \leq k}$, alors*

- $u_i \prec u_{i+1}$ *: une chaîne d'arêtes est convexe ;*
- *les rayons issus de $av(u_i)$ et de direction u_i sont stoppés par un même segment de droite O de la scène, excepté peut-être celui issu de la dernière arête u_k où alors $av(u_k)$ est une extrémité de O ;*
- *l'espace (ouvert) délimité par la demi-droite issue de u_0, par la chaîne et par O est vide de sommets.*

Démonstration. Il suffit d'appliquer la propriété 2.7 à chaque paire d'arêtes consécutives de la scène. □

Nous utilisons ces chaînes d'arêtes pour mettre à jour l'arbre d'horizon :

Proposition 2.13 *Après le passage de l'arête $u = (p, q)$, la nouvelle arête u^s partant de p de l'arbre d'horizon est l'arête issue de p et tangente à la chaîne d'arêtes se terminant par*

– *l'arête de plus petite pente située au-dessus de* (q^-, q) *si* q^- *est au-dessus de* (p, q) *;*
– *l'arête sœur gauche de* $e^s(q)$ *sinon.*

Démonstration. Toute arête de $\mathcal{E}$ arrivant en q^- dans le premier cas, ou en $av(e^s(q))$ dans le deuxième cas, et située en dessous de l'enveloppe convexe coupe forcément u, et donc ne peut appartenir à l'arbre d'horizon. L'enveloppe convexe est donc bien la partie droite de la chaîne d'arête considérée. □

Nous balayons les sommets r de la chaîne d'arêtes à partir de u_0 jusqu'à ce que nous trouvions le point de tangence avec cette chaîne (on a alors $e^s(r) \succ (p, r)$) ou que nous soyons arrivé en fin de chaîne (figure 2.8).

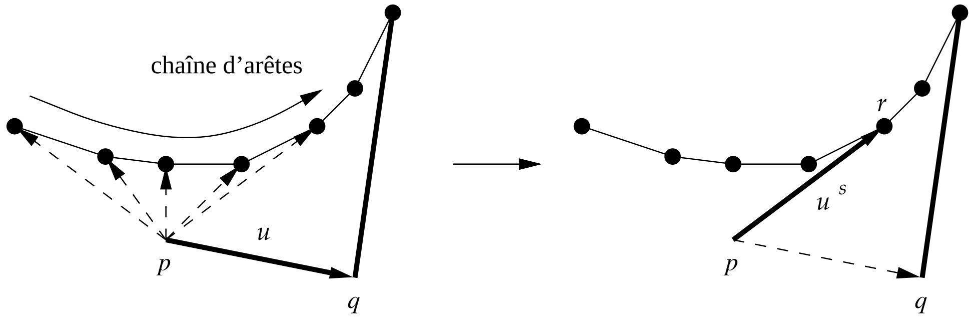

FIG. 2.8 – *Balayage d'une chaîne d'arêtes.*

Une fois le point r de tangence trouvé, la mise à jour des chaînes est directe : la chaîne est coupée en deux au niveau de r, ce qui donne une chaîne gauche et une chaîne droite, et l'arête u^s est ajoutée au début de la chaîne droite.

2.1.5 Complexité du balayage du graphe de visibilité

Nous étudions ici la complexité de notre algorithme pour balayer le graphe de visibilité.

Représentation des données

Chaque sommet p de la scène a un pointeur vers l'unique arête de l'arbre d'horizon qui en part et un pointeur sur l'arête de plus petite pente qui y arrive. Pour traiter le troisième cas de passage, chaque sommet a aussi un pointeur sur l'arête arrivante de plus petite pente supérieure à (p, p^+).

Chaque arête de l'arbre d'horizon a des pointeurs sur ses extrémités, un pointeur vers sa sœur droite et sa sœur gauche. Son père est en fait l'arête partant de son sommet droit. Les chaînes d'arêtes ne font pas l'objet d'une structure de données à part : seulement un double pointeur entre la première et la dernière arête de la chaîne.

Après un tri des sommets des polygones, la construction des arbres d'horizon et de $\min_{\prec} R_0$ se fait en temps linéaire $O(n)$.

Lors du passage d'une arête u, une fois la nouvelle extrémité de u^s trouvée, la mise à jour des différentes structures de données (arbres et chaînes d'arêtes) se fait en temps constant. Distinguer les quatre cas et trouver le cas échéant la chaîne d'arêtes à balayer se fait aussi en temps constant.

Il nous reste à calculer le coût des balayages des chaînes d'arêtes.

Balayage des sommets dans les chaînes d'arêtes

A priori, dans le cas le pire n sommets peuvent être balayés dans une chaîne, soit un coût $O(mn)$ pour le calcul du graphe de visibilité. Cependant, nous montrons dans le reste de cette partie que le nombre total des sommets balayés dans les chaînes est seulement en $O(m)$.

Nous montrons d'abord que les sommets balayés peuvent être reliés aux arêtes du graphe de visibilité :

Proposition 2.14 *Soit r un sommet d'une chaîne d'arête balayé lors de la mise à jour de l'arête u de l'arbre d'horizon. Alors l'arête $(ar(u), r)$ est une arête du graphe de visibilité.*

Démonstration. L'espace compris entre la demi-droite issue de u_0 (première arête de la chaîne), la chaîne et le segment vu O est vide. Cependant ce n'est pas suffisant pour montrer la proposition, car l'arête u est située strictement en-dessous de la demi-droite issue de u_0. Cependant l'espace de balayage compris entre la chaîne, le segment de droite $[ar(u_0), ar(u)]$ et la demi-droite issue de u et O est lui aussi vide (figure 2.9). En effet s'il y avait des sommets dans cet espace, alors il y aurait parmi ces sommets un sommet s tel que $(s, av(u))$ soit une arête du graphe de visibilité. Comme $(s, av(u)) \prec_{av(u)} u$, et comme les arêtes de l'arbre d'horizon ne peuvent se couper, l'extrémité de $e^s(s)$ est sur la chaîne d'arêtes, ce qui est impossible d'après le choix de cette chaîne.
Cet espace étant vide, tous les sommets balayés sont vus de $ar(u)$ et les arêtes correspondantes appartiennent au graphe de visibilité. □

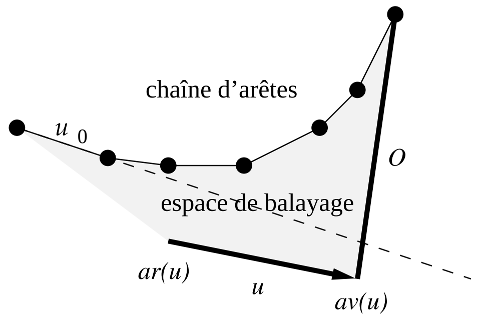

FIG. 2.9 – *Vacuité de l'espace de balayage.*

La complexité du balayage des chaînes se réduit déjà à $O(m^2)$. Nous montrons maintenant grâce à un schéma de comptage, que chaque arête du graphe de visibilité n'est en fait prise en compte seulement un nombre constant de fois lors des balayages des chaînes.

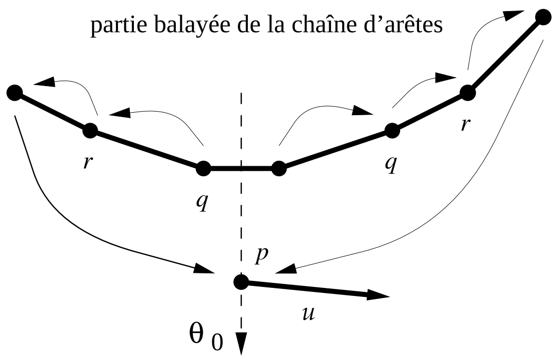

FIG. 2.10 – *Schéma de comptage : report des charges.*

Lors de la mise à jour de l'arête u de l'arbre d'horizon, considérons la partie balayée de la chaîne d'arêtes. La droite de direction θ_0 (début du balayage) passant par $ar(u)$ sépare les sommets de cette partie de chaîne en deux ensembles. Pour les sommets à droite, nous reportons la charge de balayage d'un sommet q sur le sommet suivant r de la chaîne. Pour les sommets à gauche, nous reportons la charge d'un sommet q sur le sommet précédent r de la chaîne. Enfin nous reportons la charge des deux sommets extrêmes de la partie de chaîne sur $ar(u)$ (figure 2.10).

La charge due aux sommets extrêmes est de $2m$. Nous montrons maintenant que si un sommet est chargé lors de la mise à jour d'une arête partant de p, alors il ne peut être chargé qu'une seule fois pendant le balayage entre θ_0 et $\theta_0 + \pi$ lors des mises à jour des arêtes partant de p.

Soit r un sommet chargé lors du balayage d'une partie droite de chaîne lors de la mise à jour d'une arête issue de p. Considérons la dernière fois où cette situation se produit et soit alors q le sommet qui charge r. Comme l'espace qui vient d'être balayé est vide (voir la démonstration de la proposition 2.14), avant cet événement toute arête issue de q coupe (p, r), et donc r est « caché » de p et ne peut être chargé lors de la mise à jour d'une arête issue de p.

Supposons maintenant que le sommet r est chargé lors du balayage d'une partie gauche de chaîne lors de la mise à jour d'une arête issue de p. Considérons la première fois où cette situation se produit et soit alors q le sommet qui charge r. Comme l'arête (r, q) est inférieure à la nouvelle arête u^s, elle est passée avant u^s. Si r était chargé ultérieurement par un sommet q' lors de la mise à jour d'une arête issue de p, alors r se retrouverait dans l'espace de balayage, ce qui est contradictoire avec le fait que cet espace est vide. r ne peut donc plus être chargé lors de la mise à jour d'une arête issue de p.

Lors de l'algorithme, un sommet r ne peut être chargé qu'une fois à cause de la mise à jour d'une arête issue de p, et (p, r) est une arête du graphe de visibilité. Le nombre total de charges est donc $O(m)$.

L'algorithme décrit, est donc un algorithme optimal de balayage du graphe de visibilité, qui s'exécute en temps $O(n \log n + m)$ en utilisant un espace de travail $O(n)$.

2.2 Résultats expérimentaux

Nous avons programmé cet algorithme et nous l'avons comparé expérimentalement à deux autres algorithmes de balayage de graphe de visibilité : l'algorithme d' Overmars et Welzl [OW88] qui calcule le graphe de visibilité d'une scène de segments de droite, et l'algorithme de Pocchiola et Vegter [PV93] qui calcule le graphe de visibilité d'une scène d'objets courbes quelconques, que nous avons adapté ici pour des scènes de segments.

Ces deux algorithmes, les seuls à notre disposition pour les tests, ont tous deux un temps de calcul en $O(n \log n + m \log n)$ et utilisent une queue de priorité, que nous avons implémentée avec un tas (en utilisant un tableau).

Les trois algorithmes utilisent comme opération élémentaire le calcul de la position relative de trois points. Les autres opérations sont de simples mises à jour de pointeurs, d'importance comparable dans les trois algorithmes.

La comparaison de temps d'exécution dans ce cadre n'a pas beaucoup de sens : les différences de temps dépendraient plus des différences d'optimisations de programmation de bas niveau (qui peuvent accélérer de 50% un programme par rapport à un programme non optimisé). Nous avons alors mesuré la complexité « algorithmique » des trois algorithmes : nous avons compté le nombre d'opérations élémentaires effectuées lors du calcul du graphe de visibilité (sans prendre en compte la phase d'initialisation des diverses structures de données).

Pour effectuer les tests, nous avons généré aléatoirement des scènes de segments. Pour chaque segment $[p_1, p_2]$, le premier point p_1 est tiré au hasard dans le carré centré en $(0, 0)$ et de côté $2a$. Le second point est tiré au hasard dans le carré centré en p_1 et de côté $2b$. Nous avons effectué les tirages en utilisant une loi de probabilité uniforme. Pour un nombre de segments n fixé, nous avons fait varier la taille du graphe de visibilité en jouant sur les paramètres a et b. Par exemple, les scènes générées avec b pas trop petit par rapport à a ont un graphe de visibilité peu dense ($m = O(n)$), alors que les scènes générées avec b très petit devant a ont un graphe de visibilité dense ($m = O(n^2)$).

Pour une valeur fixée de n nous avons généré 500 scènes aléatoires de façon à obtenir des graphes de visibilité à peu près uniformément répartis entre graphes peu denses et graphes denses. La complexité du temps de calcul des graphes en fonction de leur taille devrait alors être une droite d'équation $complexite = c(n) * m + m_0$. Les mesures obtenues sont effectivement réparties selon une droite comme le montre la figure 2.11 pour des scènes de $n = 500$ segments. Nous avons alors calculé la pente $c(n)$ de la droite en utilisant une régression linéaire.

Nous avons calculé cette constante de complexité $c(n)$ pour $n = 100, 200, \ldots, 1000$ et la figure 2.12 montre la courbe de complexité résultante où nous avons tracé les points $c(n)$ en fonction de n.

Nous observons que les algorithmes d'Overmars et Welzl [OW88] et de Pocchiola et Vegter [PV93] sont très proches et la nature logarithmique de leur complexité en $O(m \log n)$ est visible (les courbes d'interpolations tracées pour ces deux algorithmes sont des régressions logarithmiques).

L'algorithme de balayage topologique est meilleur et sa complexité est effectivement indépendante de n. La complexité d'implémentation du programme étant de même or-

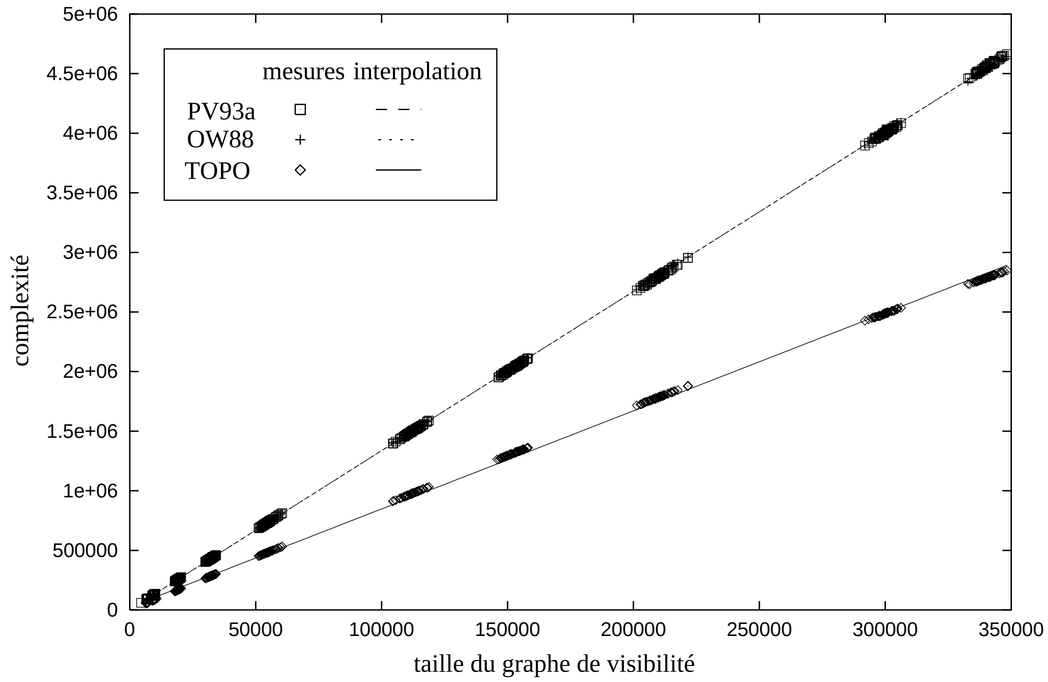

FIG. 2.11 – *Mesure de la complexité pour* $n = 500$.

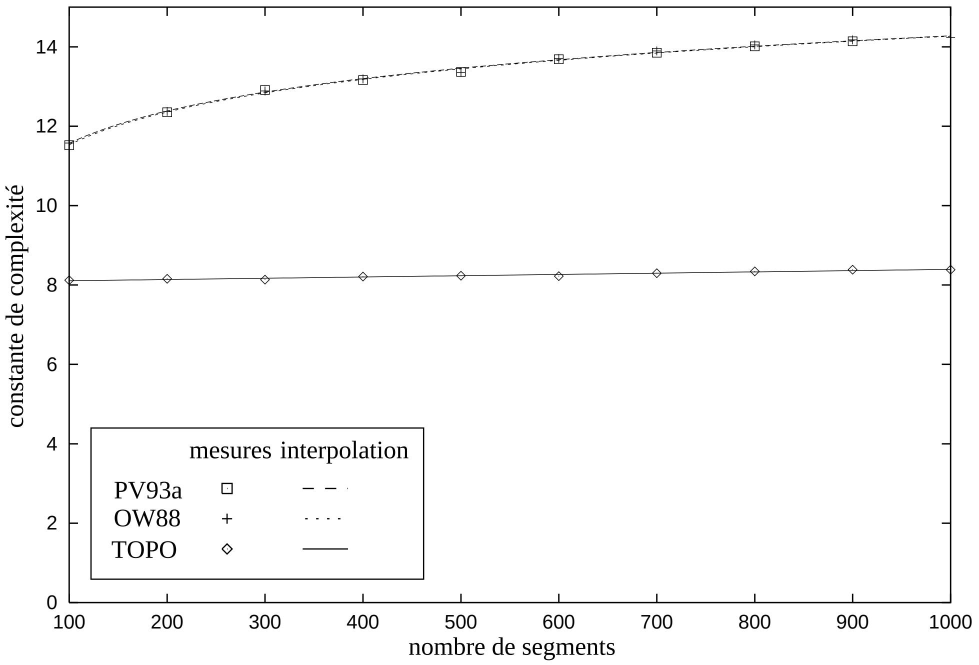

FIG. 2.12 – *Constante de complexité en fonction de* n.

dre que celles des deux autres programmes (nous avons même réutilisé une partie du code que nous avons écrit pour l'algorithme d'Overmars et Welzl dans l'implémentation de l'algorithme de balayage topologique), l'algorithme est aussi plus efficace en pratique.

2.3 Synthèse

Dans cette partie nous revenons sur les liens existants entre les algorithmes utilisant les arrangements de droites, l'algorithme d'Overmars et Welzl [OW88], le complexe de visibilité et notre algorithme de balayage topologique.

Comme nous l'avons déjà indiqué, le cadre combinatoire formel définissant le balayage à l'aide de filtres doit beaucoup à la formalisation qu'utilisent Pocchiola et Vegter [PV96a] dans la description de leur algorithme de calcul optimal de graphe de visibilité d'objets convexes.

Liens avec les arrangements de droites

On peut dire que notre algorithme de balayage de graphe de visibilité est au complexe de visibilité, ce que l'algorithme de balayage topologique d'Edelsbrunner et Guibas [EG86] est aux arrangements de droites.

Le balayage topologique du graphe de visibilité peut être vu comme le balayage topologique du squelette du complexe de visibilité dans l'espace dual. Puisque le squelette du complexe se projette sur l'arrangement de droites dual des points de la scènes, nous remarquons des analogies avec le balayage d'arrangements de droites. Cependant les différences entre les droites et les segments libres maximaux créent des différences fondamentales que nous devons prendre en compte ici. Ainsi, avec le formalisme utilisé dans ce chapitre, dans l'algorithme d'Edelsbrunner et Guibas [EG86] l'ensemble des arêtes à passer est simplement : $\min_{\prec} R = \mathcal{E}^s \cap \mathcal{E}^i$. Ici, comme nous l'avons vu, la condition doit être renforcée du fait d'une certaine indépendance entre arêtes dont les extrémités sont mutuellement cachées (figure 2.13).

Le schéma de comptage présenté dans la preuve de complexité est une adaptation de celui utilisé par Edelsbrunner et Guibas [EG89], transposé des arrangements de droites à la scène polygonale. Mais là aussi des précautions ont du être prises à cause de la nature différente du complexe.

Liens avec l'algorithme d'Overmars et Welzl [OW88]

Comme l'algorithme d'Overmars et Welzl [OW88] prend directement en compte les visibilités, il se démarque des algorithmes de balayage d'arrangements de droites. Cet algorithme reste cependant prisonnier de certaines limitations, notamment du sommet factice à l'infini correspondant à la droite verticale factice ajoutée à la fin de l'arrangement pour arrêter le balayage.

Ici, bien que nous ayons indiqué à chaque fois les rapports entre la situation dans la scène et dans l'espace dual, notre algorithme n'a pas été envisagé comme une « dédualisation » d'un algorithme lié à l'arrangement dual de droites. Le fait de considérer l'ensemble infini des arêtes orientées du graphe de visibilité et de faire abstraction de la relation de dualité utilisée nous a permis par exemple de faire disparaître le point factice. Nous pouvons alors faire un balayage infini, ce qui nous permet notamment

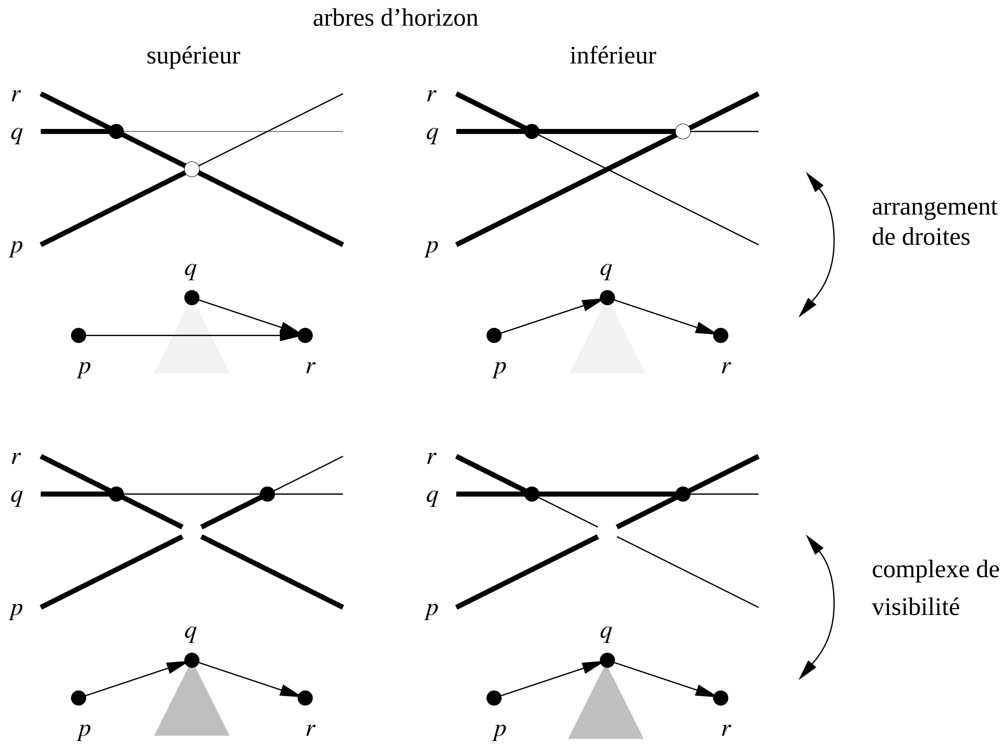

FIG. 2.13 – *Différences dues à la prise en compte de la visibilité.*

de calculer le graphe de visibilité de polygones avec trous (trous pouvant contenir eux-mêmes d'autres polygones...), ce qui n'était pas possible avant.

Retour à l'espace dual

Chaque point de vue apporte ses propres caractéristiques pour résoudre un même problème, ce qui permet d'avoir plus d'outils à sa disposition.

La vision duale permet elle aussi de rajouter des outils :

Proposition 2.15 *Si dans la représentation duale du complexe, on peut tracer un segment de droite d'une arête p^* à une arête q^*, tel que la pente de ce segment soit supérieure (resp. inférieure) à celles des deux arêtes et que ce segment ne coupe pas d'autres arêtes de pente supérieure (resp. inférieure), alors dans la scène, les deux points p et q correspondant aux deux arêtes sont mutuellement visibles.*

Démonstration. La droite support du segment est la droite duale d'un point r. Le segment représente des demi-droites pivotant autour de r, depuis p jusqu'à q. Si dans le complexe, ce segment ne rencontre pas d'arête de pente plus grande, cela signifie qu'il n'y a pas de points de la scène dans le cône $(\overrightarrow{rp}, \overrightarrow{rq})$ (restreint aux régions de visibilité correspondantes du complexe), et donc les points p et q sont mutuellement visibles (figure 2.14). $\square$

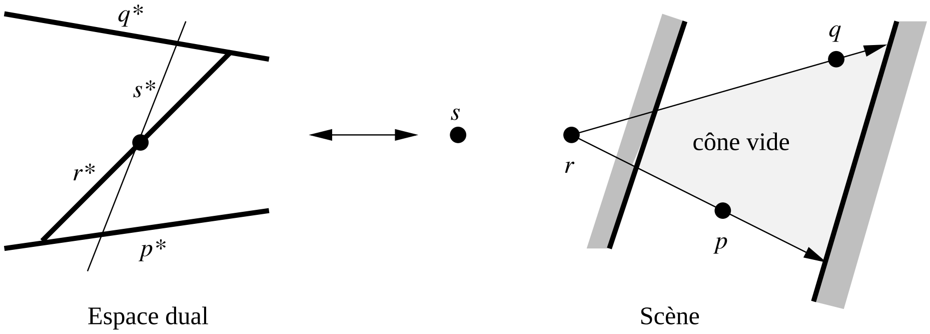

FIG. 2.14 – *Visibilité démontrée dans l'espace dual.*

2.4 Construction du complexe de visibilité

Nous montrons ici comment construire le complexe de visibilité en utilisant l'algorithme de balayage topologique.

Nous avons vu que l'algorithme de balayage topologique du graphe de visibilité effectuait en fait un balayage du squelette du complexe de visibilité. Grâce à la cohérence de ce balayage, nous pouvons construire les éléments du complexe.

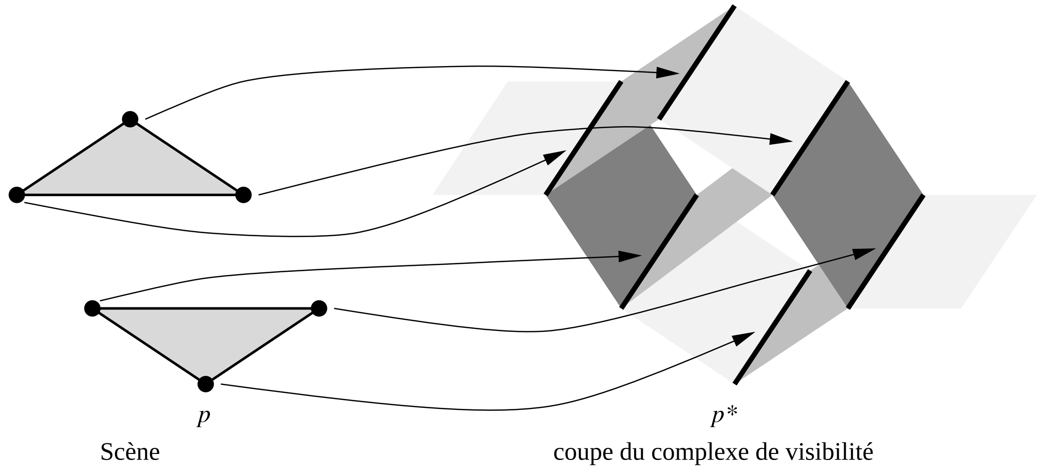

FIG. 2.15 – *Association d'arêtes de la coupe du complexe aux sommets de la scène.*

Lors de ce balayage, nous associons à chaque sommet p de la scène l'arête du complexe de visibilité « coupée » par la ligne de balayage et dont le support est la courbe duale p^* (figure 2.15). Plus précisément, chaque arête du graphe de visibilité correspond à un sommet du complexe. Nous associons au sommet p l'arête du complexe telle que l'arête correspondant à son sommet gauche a déjà été balayée et l'arête associée à son sommet droit appartient au moins à l'un des deux arbres d'horizon.

Les faces du complexe ne sont pas mises en correspondance directe avec un élément de la scène : nous y accédons par l'intermédiaire des arêtes du complexe déjà créées.

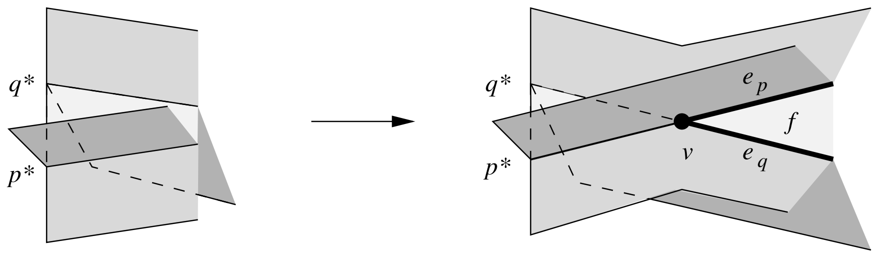

FIG. 2.16 – *Exemple de construction du complexe lors du passage d'une arête du graphe de visibilité.*

Le complexe est construit au fur et à mesure lors du passage d'une arête (p, q) du graphe de visibilité comme dans l'exemple de la figure 2.16. Ce passage indique le passage d'un sommet v dans le complexe. À ce moment là, les deux arêtes gauches incidentes au sommet et les faces incidentes à ces arêtes sont déjà créées, mais leur construction est incomplète. Lors du passage, nous créons le sommet v et mettons à jour les relations d'incidence entre le sommet et ses deux arêtes gauches associées à p et q. Nous créons les deux arêtes droites e_p et e_q de v, mettons à jour les relations d'incidences entre ces éléments et associons les arêtes e_p et e_q respectivement aux points p et q. Enfin, nous créons la face f ≪ ouverte ≫ par v (c'est-à-dire la face dont v est le sommet extrême gauche) et mettons à jour les relations d'incidences entre les nouvelles arêtes e_p et e_q et les faces incidentes aux arêtes droites de v (dont f fait maintenant partie).

La configuration locale du complexe autour d'un sommet (nombre de faces et relations d'incidences entre ces faces et les arêtes incidentes au sommet) dépend de la configuration géométrique de la scène autour des sommets de l'arête du graphe de visibilité correspondante. Comme nous l'avons vu au chapitre précédent, il y a 17 configurations différentes à traiter. Chaque configuration est uniquement déterminée par les types des arêtes incidentes gauches du sommet et finalement, la construction du ≪ morceau ≫ de complexe autour du sommet passé se fait en temps $O(1)$.

Il ne faut pas oublier de calculer les étiquettes des éléments nouvellement créés du complexe. Mais là aussi, grâce à la cohérence du balayage les visibilités sont mises à jour en temps constant : elles ne sont modifiées qu'en fonction de la configuration géométrique locale de la scène.

Grâce au balayage topologique, le complexe peut être construit en temps optimal $O(n \log n + m)$ et servir ensuite a effectuer des calculs de visibilité comme nous le montrons dans le chapitre suivant.