Département
Informatique
Fondamentale

Quelques remarques sur l'algorithme A*

M. Habib, G. Simonet

R. R. Juin 1991 - 1

Rapport de Recherche

LIRMM, CNRS et Université de Montpellier II

860, rue de Saint-Priest F34100, Montpellier, France Tel: (33) 67 63 04 60 Fax: (33) 6754 72 76 email: dif@crim.fr

Quelques remarques sur l'algorithme A*

M. Habib, G. Simonet

Dépt. d'Informatique Fondamentale
(LIRMM, CNRS et Université de Montpellier II)
860, rue de Saint-Priest
34100, Montpellier, France.
email: habib@crim.fr

Résumé: Dans cette étude, il s'agit d'étudier précisemment les propriétés du célèbre algorithme d'intelligence artificielle A* introduit par Hart, Nilsson et Raphael (1968), afin de mieux le comprendre et d'être en mesure de le généraliser. L'originalité de ce travail tient à ce que nous exhibons et démontrons (sans formalisme outrancier) les principaux invariants de l'algorithme A*, essayant par là de mieux capter les propriétés combinatoires de ce très intéressant algorithme.

Ainsi nous obtenons une propriété (théorème 1) qui ouvre la voie à de nouvelles implémentations de A*.

Enfin nous concluons en présentant dans ce contexte les principaux résultats connus (admissibilité, consistance, variantes), en essayant à chaque fois de mettre en évidence les propriétés générales sous-jacentes (cf. théorème 2, proposition 4).

Mots clés: algorithmes de recherche de plus courts chemins dans un graphe, parcours heuristiques : A*, B et algorithme de Dijkstra, information heuristique, admissibilité, consistance, invariants et preuve d'algorithmes, complexité.

I INTRODUCTION ET PRESENTATION DE L'ALGORITHME

L'algorithme A* a été introduit dès les débuts de l'intelligence artificielle par Hart, Nilsson et Raphael (1968) afin de résoudre des problèmes de plus courts chemins dans des graphes valués de taille tellement grande qu'il y est exclu d'utiliser les algorithmes classiques de plus courts chemins tels celui de Dijkstra (1959). L'algorithme A* est un algorithme de parcours de graphe qui introduit une dimension heuristique au parcours, en cherchant à limiter l'ensemble des chemins à considérer. Bien entendu ceci peut se faire au détriment de l'optimum (i.e. l'algorithme A* ne donne pas toujours un chemin optimal). Les applications de A* sont très nombreuses et vont de la reconnaissances des formes Martelli (1972, 1976), Bertier et Habib (1983), à la navigation de robot mobile, sont décrites dans Farreny et Ghallab (1987) et justifient donc une analyse précise de l'algorithme.

L'algorithme A* qui sera décrit en détail (la Figure 1, ci-après) est un très bel objet combinatoire et nous allons travailler sur les invariants et propriétés de ce programme.

Les données du problème:

Pour un sommet x d'un graphe G, on appelle **successeurs** de x (resp. **descendants** de x), les sommets extrémités des arcs (resp. chemins) issus de x.

On considère un graphe G=(X,U) orienté, localement fini (chaque sommet admet un nombre fini de successeurs (mais le graphe lui-même peut être infini, c'est à dire posséder un nombre infini de sommets). Cependant les chemins que nous considérons dans ce travail sont **finis**.

De plus G est muni d'une valuation $v: U \rightarrow \mathbf{R}$. La valuation d'un chemin μ de G, notée $v(\mu)$ est la somme des valuations des arcs qui le composent. On distinguera cette valuation de la longueur du chemin (nombre d'arcs du chemin). On définit la distance d'un sommet à un autre comme suit:

$d(x,y)= \inf_{\mu \text{ chemin de } x \text{ à } y} (v(\mu))$. (Cette définition à l'aide d'une borne inférieure au lieu d'un minimum, permet d'inclure le cas où d(x,y) n'est pas atteinte sur un chemin, mais correspond à une limite sur un ensemble de chemins). On étend naturellement cette définition à la distance d'un sommet à un ensemble $Y \subseteq X$ de sommets:

$d(x,Y)= \inf_{y\in Y}(d(x,y))$.

Etant donné un sommet origine s et un ensemble P de sommets destination, il s'agit de trouver le plus court chemin dans G allant de s à un sommet de P.

On suppose en outre connue ou calculable en tout sommet x du graphe, la valeur h(x) estimation de la distance entre x et P. Cette fonction dite "heuristique" est essentielle pour A*, car elle va intervenir dans les choix successifs des sommets à explorer.

Données: un graphe G=(X,U), s ∈ X et P ⊆ X, une valuation v: U → **R**, **h**: X → **R**.
Résultat: un chemin de s à P, de valuation minimale, s'il existe un chemin de s à P.

```
début
{Début Initialisations}
g(s) ← 0;
Ouverts ← {s}; Fermés ← Ø;
pred(s) ← NIL;
{Fin initialisations}

{Les ensembles Ouverts et Fermés servent à marquer les
sommets lors du parcours du graphe: un sommet ouvert est un
sommet à explorer, tandis qu'un sommet fermé a déjà été
exploré}
tant que Vide(Ouverts)= FAUX
  faire     x ← Choix(Ouverts, f, g, h, P);
            Ajout(x, Fermés);
            Retrait(x, Ouverts);
       si x ∈ P alors STOP
       {on a trouvé un chemin de s à P}

       Sinon {explorer x}
       pour tout successeur y de x
       faire
       cas
       1. y ∈ Ouverts
            si g(y)>g(x)+v(xy) alors
            faire
            g(y)←g(x)+v(xy);
            pred(y) ← x;
       2. y ∈ Fermés
            si g(y)>g(x)+v(xy) alors
            faire
            Ajout(y, Ouverts);
            Retrait(y, Fermés);
            g(y)←g(x)+v(xy);
            pred(y) ← x;
       3. y ∉ Fermés ∪ Ouverts
            faire
            Ajout(y, Ouverts); g(y)←g(x)+v(xy);
            pred(y) ← x;
       Fincas
fintantque
{Il n'existe pas de chemin de s à un sommet de P}
fin
```

Figure 1

Commentaires:

Dans cet algorithme on suppose que la fonction Choix(A, f, g, h, P) où A et P sont des ensembles de sommets du graphe et f une fonction définie sur A à partir des fonctions g et h (souvent f=g+h), admet pour résultat un sommet de A, qui minimise la fonction f. En cas d'égalité on suppose que la fonction choix privilégie les éléments de P.

La valeur pred(x) définie pour tout sommet ouvert ou fermé, va permettre de construire effectivement un chemin de s à P, lorsque l'algorithme en trouve un. Afin de pouvoir traiter de gros graphes, pred(x) n'est volontairement pas initialisée sur tous les sommets du graphe, et donc cette valeur n'a de signification que pour les sommets qui ont été ouverts par l'algorithme.

Analyse en complexité:

En première analyse, la complexité de l'algorithme précédent dépend du nombre total I d'explorations de sommets par A*. Un même sommet pouvant bien entendu être ouvert plusieurs fois et donc exploré plusieurs fois (cf.exemple Figure 2). Ce nombre I est souvent assimilé à la complexité de l'algorithme. Cependant on peut distinguer une complexité structurelle (liée au parcours du graphe lors de l'exploration des sommets) s'exprimant comme suit:

$$E = \sum_{x \text{ exploré par } A^*} \delta^+(x) \times \text{nbrex}(x)$$

où $\delta^+(x)$ et nbrex(x) représentent respectivement le degré extérieur (nombre d'arcs sortant) et le nombre d'explorations du sommet x.

La deuxième complexité, un peu masquée dans la présentation ci-dessus de l'algorithme dans la figure 1, tient à la gestion des ensembles des Ouverts, des Fermés et à l'implémentation de la fonction Choix. Ainsi la complexité de l'algorithme va dépendre de la structure de données utilisée pour la mise en oeuvre des ensembles Ouverts et Fermés. Pour Fermés il suffit d'étiqueter les sommets. Par contre pour Ouverts la fonction Choix doit prendre un sommet de cet ensemble qui minimise la fonction f. Il faut donc réaliser le mieux possible les opérations: **Insérer-élément, Modifier-valeur** et **Détruire-minimum**. Pour ce faire il est classique d'utiliser une queue de priorité, pour laquelle ces opérations se font en O(logC), où C représente la taille maximale de l'ensemble des sommets ouverts du graphe, au cours de l'algorithme. On obtient alors une complexité globale majorée par O(I.logC + E.logC). (Cette majoration est grossière car on y suppose que l'on fait une mise à jour à chaque arc visité).

Mais on peut aussi utiliser une structure de données de type Fibonacci-Heap due à Fredman et Tarjan (1984) pour représenter l'ensemble des ouverts. Cette dernière structure de données permet de réaliser Détruire-Min en O(logC) et de réaliser m opérations Insérer-élément et Modifier-valeur en O(m). Ce dernier résultat utilise les notions de complexité amortie, et donne une complexité globale de O(I.logC + E).

En regroupant les deux complexités on obtient donc une complexité globale de O(I.logC + E), en utilisant les Fibonacci-Heap pour la gestion de l'ensemble des ouverts.

Dans le cas où le graphe G est fini et de taille $|X|=n$, on peut majorer cette expression par O(I.logn + E).

Nous présentons tout d'abord au II, les principaux invariants généraux de A* (ne nécessitant aucune hypothèse sur la fonction de choix) G1-G10. Nous exhibons au passage une propriété structurelle intéressante de l'ensemble des ouverts (théorème 1) qui permet de définir une variante plus efficace de A*.

Enfin au III, nous présentons dans ce cadre les propriétés classiques de l'algorithme A*, à savoir: admissibilité, consistance et variantes. Ce faisant nous montrons que la propriété de consistance permet de se ramener à l'algorithme de Dijkstra, et nous présentons une condition suffisante d'admissibilité qui s'applique aussi sur la variante B proposée par Martelli.

II PROPRIETES DE A*

Avant de procéder à l'analyse il nous faut introduire quelques notations:

Nous indicerons par i ($1 \leq i \leq I$), les ensembles (Ouverts et Fermés) et les valeurs des fonctions (g, pred) pour un sommet x, ces valeurs étant prises au début de la ième itération de la boucle tantque.

On note $\text{Atteints}_i = \text{Ouverts}_i \cup \text{Fermés}_i$, et Choix_i le sommet obtenu par la fonction Choix à l'étape i.

Nous dirons que x est ouvert (resp. fermé, atteint) à l'étape i, lorsque $x \in \text{Ouverts}_i$ (resp. Fermés_i, Atteints_i). Enfin on notera d(x,P) la valuation d'un plus court chemin de x à P dans le graphe, et nous allons voir dans quelle mesure A* calcule d(s,P).

Hypothèses minimales sur le graphe:

Afin de donner un sens à l'algorithme précédent il faut faire quelques hypothèses sur le graphe G initial.

(i) G n'admet pas de circuit de valuation négative.
(ii) $\forall$ p',p" $\in$ P, d(p',p")$\geq$0.
(iii) il existe un chemin de longueur finie de s à un sommet de P.

Dans ce qui suit nous ferons ces hypothèses sur le graphe, sans les mentionner.

Nous pouvons maintenant énoncer les propriétés fondamentales de A*, les invariants de l'algorithme. Pour commencer examinons ceux qui n'utilisent aucune hypothèse sur la fonction h ou la fonction de choix.

Invariants généraux:

En simplifiant on peut comprendre l'algorithme A*, comme une seule boucle "tant que", et nous avons choisi de placer nos invariants à l'entrée de la ième itération. Les instructions d'initialisation determinent ainsi la première valeur de l'invariant de l'itération, et si l'on a réussi à établir la validité d'un invariant INV(i), à la fin de l'algorithme (à l'issue de la Ième itération) la propriété suivante est vraie:

$\{(\text{Ouverts}_I = \emptyset) \vee (p = \text{Choix}_I \text{ et } p \in P)\} \wedge \text{INV}(I)$

Invariant G1: Atteints $_i \subseteq$ Atteints $_j$, pour i<j. Si un sommet x est fermé à l'étape i, alors tous ses successeurs sont atteints à l'étape i.

Preuve: Lorsqu'un sommet est atteint, il l'est définitivement et un sommet n'est fermé qu'après son exploration. Enfin l'exploration d'un sommet permet d'atteindre tous ses successeurs.

Invariant G2: Pour tout sommet x, la fonction: $i \rightarrow g_i(x)$ est décroissante.

Preuve: Cette remarque est immédiate, car dans l'algorithme, la valeur g(x) n'est modifiée qu'avec une valeur inférieure à la précédente.

Invariant G3: Si x est un sommet atteint à l'étape i, alors il existe un chemin élémentaire μ de s à x tel que $g_i(x) = v(\mu)$ et pour tout $y \in \mu$, $y \neq x$, il existe j<i tel que y=Choix $_j$ et $g_j(y) = v(\mu[s,y])$.

Preuve: La preuve se fait par induction sur le nombre d'étapes de l'algorithme.

Pour i=1, x=s et $\mu=\{s\}$, $g_1(s)=v(\mu)=0$ et c'est trivialement vrai.

Supposons la propriété vraie, jusqu'à l'étape i, montrons qu'elle est encore vérifiée avant l'exécution de l'étape i+1. Pour ce faire on considère $x \in$ Atteints $_{i+1}$. Deux cas sont possibles:

a) $g_{i+1}(x) = g_i(x)$ et dans ce cas x était déjà atteint à l'étape i et l'on peut appliquer intégralement l'hypothèse d'induction.

b) Dans le cas contraire, posons z=Choix $_i$, nécessairement: $g_{i+1}(x) = g_i(z) + v(z,x)$. Comme z était atteint à l'étape i, il existe un chemin μ de s à z, satisfaisant les conditions de l'invariant 3. Soit μ' le chemin μ prolongé de l'arc (z,x). Montrons que μ' vérifie bien les conditions pour x.

En effet on a bien $g_{i+1}(x) = g_i(z) + v(z,x) = v(\mu) + v(z,x) = v(\mu')$. La suite de la condition est trivialement vérifiée pour μ' et x, il ne nous reste plus qu'à vérifier qu'il est bien élémentaire.

Supposons le contraire, par hypothèse d'induction, seul x peut être répété. Cependant si $x \in \mu$, il existe donc $j<i$ tel que $g_j(x) = v(\mu_{[s,x]})$ (où $\mu_{[s,x]}$ représente le sous chemin de μ allant de s à x). Avec nos hypothèses $g_{i+1}(x) < g_i(x)$, en utilisant l'invariant 2, on obtient:

$g_{i+1}(x) = v(\mu) + v(z,x) < g_i(x) \leq g_j(x) = v(\mu_{[s,x]})$, ce qui est exclu, car il n'existe pas par hypothèse de circuit de valuation négative dans le graphe.

#

De cet invariant nous pouvons déduire, en prenant la convention, $g_i(x)= +\infty$ lorsque x n'est pas atteint à l'étape i.

Invariant G3.1: Pour tout sommet $x \in X$, i, $1 \leq i \leq I$, $g_i(x) \geq d(s,x)$.

Preuve: le résultat est évident lorsque $x \notin$ Atteints $_i$ avec la convention précédente. Dans le cas contraire, il existe d'après l'invariant G3, un chemin élémentaire μ de s à x et de valeur $g_i(x)$, d'où le résultat.

#

Invariant G4: Si x est fermé à l'étape i, alors pour tout successeur y de x, on a $g_i(y) \leq g_i(x) + v(x,y)$.

Preuve: Par induction sur i. Lorsque i=1, le résultat est trivial.
Supposons la propriété vraie jusqu'à l'étape i, et procédons à une étape supplémentaire. Considérons $x \in$ Fermés $_{i+1}$ deux cas sont possibles:

a) $x \in$ Fermés $_i$, et nous avons:

$g_{i+1}(x) = g_i(x)$ car sinon x aurait été réouvert à l'issue de l'étape i.

$g_{i+1}(y) \leq g_i(y)$ (invariant G2).

$g_i(y) \leq g_i(x) + v(x,y)$ (hypothèse d'induction).

D'où l'on déduit: $g_{i+1}(y) \leq g_{i+1}(x) + v(x,y)$.

b) $x \notin$ Fermés $_i$, dans ce cas nécessairement $x =$ Choix $_i$. Il est alors vrai que:

$g_{i+1}(y) = \min(g_i(y), g_i(x) + v(x,y))$ et ainsi comme $g_{i+1}(x) = g_i(x)$, nous avons: $g_{i+1}(y) \leq g_{i+1}(x) + v(x,y)$.

#

Invariant G5: Soit μ un chemin de s à x, composé uniquement de sommets fermés à l'étape i (sauf éventuellement x), alors $g_i(x) \leq v(\mu)$.

Preuve: par induction sur la longueur k de μ.

Pour k=0, x=s, $g_i(s) = v(\mu) = 0$, et la propriété est trivialement vraie.

Considérons un chemin μ de longueur k+1 de s à x, n'utilisant que des sommets fermés à l'étape i.

Soit y le sommet à distance k de s sur μ. Par induction nous avons: $g_i(y) \leq v(\mu[s,y])$. D'après l'invariant G4, $g_i(x) \leq g_i(y) + v(y,x)$.

D'où l'on déduit $g_i(x) \leq v(\mu)$.

#

Remarque: comme le montre l'exemple de la figure 2, on n'a pas toujours l'égalité ($g_i(x) = v(\mu)$) dans l'invariant G5.

Etudions maintenant les propriétés du graphe défini par la fonction pred.

Invariant G6: Soient $x,y \in$ Atteints $_i$, $1 \leq i \leq I$, si $x=pred_i(y)$, alors :
soit $g_i(y) = g_i(x) + v(x,y)$ si $x \in$ Fermés $_i$
soit $g_i(y) > g_i(x) + v(x,y)$ si $x \in$ Ouverts $_i$

Preuve: remarquons que $x=pred_i(y)$ implique que xy est un arc du graphe G initial. La preuve se fait par récurrence sur i.

Pour i=1, comme pred(s)=NIL, le résultat est trivialement vrai.

Supposons la propriété vraie jusqu'à l'ordre i, $1 \leq i < I$, et montrons qu'elle se conserve à l'ordre i+1.

Posons $x_0=$Choix $_i$, et considérons $x,y \in$ Atteints $_{i+1}$, tels que $x=pred_{i+1}(y)$.

• si $x=x_0$, alors :

a) si $x=pred_i(y)$, x était ouvert à l'étape i et donc par hypothèse d'induction $g_i(y) > g_i(x) + v(x,y)$, g(y) est modifié à la i+1 ème étape et

$g_{i+1}(y) = g_i(x) + v(x,y)$, comme $g_{i+1}(x)=g_i(x)$, ce cas est réglé car $x \in$ Fermés $_{i+1}$.

b) si $x \neq pred_i(y)$, comme $x=pred_{i+1}(y)$, nécessairement $g(y)$ a été modifié à l'étape i+1 et l'on termine comme en a).

• $x \neq x_0$, alors nécessairement $x=pred_i(y)$ et $g_{i+1}(y)=g_i(y)$.

a) supposons $g(x)$ ait été modifié à l'étape i+1. Alors $g_{i+1}(x) < g_i(x)$ et $x \in$ Ouverts $_{i+1}$. Par hypothèse d'induction $g_i(y) \geq g_i(x) + v(x,y)$ (ici on ne précise pas si $x \in$ Fermés $_i$ ou non). On en déduit donc $g_{i+1}(y) > g_{i+1}(x) + v(x,y)$.

b) Dans le cas contraire, $g_{i+1}(x)=g_i(x)$. x reste dans l'état (ouvert ou fermé), qu'il avait à l'étape i, et comme $g_{i+1}(y)=g_i(y)$, nous pouvons conclure grâce à l'hypothèse d'induction.

#

Invariant G7: Soient $x,y \in$ Atteints $_i$, $1 \leq i \leq I$, si $x=pred_i(y)$, alors : $g_i(y) \geq g_i(x) + v(x,y)$

Preuve: immédiat, d'après l'invariant G6.

#

Rappellons sans démonstration une caractérisation bien connue des arborescences qui nous sera très utile dans la preuve de l'invariant G8:
Un graphe orienté $A=(Y,V)$ est une arborescence de racine s si et seulement si A est connexe et $\delta_A^-(s)=0$ et $\forall x \in Y\text{-}s$, $\delta_A^-(x) =1$. (Où $\delta_A^-(x)$ représente le degré intérieur du sommet x dans le graphe A).

A tout sommet $x \in Y$, on peut associer le chemin unique de l'arborescence A, qui va de s à x. Notons $A[s,x]$ cet unique chemin.

Invariant G8: $\forall$ i, $1 \leq i \leq I$, $A_i=($Atteints $_i$, $V_i)$ où $V_i=\{xy \mid pred_i(y)=x\}$ est une arborescence de racine s.

Preuve: par induction sur i. Lorsque i=1, la propriété est trivialement vraie, l'arborescence est réduite à sa racine s. Supposons la propriété vraie jusqu'à la ième itération et vérifions qu'elle se conserve lorsqu'à la ième itération on explore un sommet x_0. Par hypothèse d'induction, il existe un chemin unique de s à x_0 dans l'arborescence A_i, noté $A_i[s,x_0]$.

Les opérations effectuées vont être du type $pred(y) \leftarrow x_0$. Vu la caractérisation précédente, la condition sur les degrés étant manifestement vérifiée (car pred(x) ne contient qu'une valeur pour un sommet x donné), il suffit de vérifier que la connexité est préservée lorsqu'on modifie l'arborescence.

Soit K_i le nombre d'opérations pred(y) $\leftarrow x_0$, réalisées au cours de l'exploration de x_0. Notons $A_{i,j}$, $0 \leq j \leq K_i$, le graphe A_i après la jème opération. Ainsi $A_{i,0} = A_i, \ldots, A_{i,K_i} = A_{i+1}$. Montrons par induction sur j la propriété suivante:

P_j : $A_{i,j}$ est une arborescence de racine s et $A_{i,j}[s,x_0]=A_i[s,x_0]$

Pour j=0, comme $A_{i,0}= A_i$, la propriété P_0 est vraie grâce à l'hypothèse d'induction sur i. Supposons P_j vraie, pour $0 \leq j < K_i$, et montrons qu'après une j+1ème modification du type pred(y) $\leftarrow x_0$, P_{j+1} est vraie.

a) Si $y \notin$ Atteints $_i$, lorsqu'on ouvre le sommet y, l'arc x_0y permet d'ajouter le sommet y à l'arborescence tout en préservant la connexité. On peut dans ce cas conclure, grâce à la caractérisation ci-dessus des arborescences et aux hypothèses d'induction sur j.

b) Si $y \in$ Atteints $_i$. $A_{i,j+1}$ se déduit de $A_{i,j}$ en remplaçant l'arc ($pred_i$(y), y) par (x_0, y). La suppression de l'arc ($pred_i$(y), y) disconnecte l'arborescence $A_{i,j}$ en 2 composantes connexes qui sont: $A_{i,j}$ - Desc $_{A_{i,j}}$(y) et Desc $_{A_{i,j}}$(y). L'ajout de l'arc (x_0, y) reconnecte ces deux composantes si et seulement si $x_0 \notin$ Desc $_{A_{i,j}}$(y).

Il suffit donc de vérifier que y n'est pas un antécédent de x_0 dans l'arborescence $A_{i,j}$. Pour ce faire supposons que y appartienne au chemin qui va de s à x_0 dans $A_{i,j}$, c'est à dire dans A_i, d'après l'hypothèse de récurrence P_j. Notons $\mu=[y=y_0, y_1, \ldots, y_k=x_0]$ le chemin de y à x_0 dans A_i (il se peut que y=s), d'après l'invariant G7, nous avons:

$g_i(x_0) \geq g_i(y_{k-1}) + v(y_{k-1},y_k) \geq \Sigma\ v(y_j,y_{j+1}) + g_i(y) = v(\mu) + g_i(y)$.

Or $g_i(y) > g_i(x_0) + v(x_0,y)$, d'où $g_i(x_0) > v(\mu) + v(x_0,y) + g_i(x_0)$

ce qui entraine $v(\mu)+v(x_0,y)=v(\sigma) < 0$ (où σ est le circuit obtenu en fermant le chemin μ par l'arc x_0y). Ainsi σ est un circuit de valuation négative, ce qui est contraire aux hypothèses faites sur le graphe G, d'où la contradiction.

Donc $y \notin A_{i,j}[s,x_0]$, et grâce à la préservation de la connexité (vérifiée ci-dessus) et à la caractérisation ci-dessus des arborescences, $A_{i,j+1}$ est bien une arborescence de racine s.

Ainsi nous avons $A_{i,j+1}[s,x_0]=A_{i,j}[s,x_0]=A_i[s,x_0]$ la dernière égalité provient de l'hypothèse d'induction sur j.

Ainsi dans les deux cas a et b, nous avons obtenu P_{j+1}.

#

A chaque étape i de l'algorithme, pour tout sommet $x \in$ Atteints $_i$, on peut donc considérer l'unique chemin $A_i[s,x]$ de l'arborescence A_i, qui va de s à x.

Invariant G9: $\forall$ i, $1 \leq i \leq I$, $\forall$ $x \in$ Atteints $_i$, $g_i(x) \geq v(A_i[s,x])$

Preuve: La preuve se fait par induction sur la longueur k du chemin $A_i[s,x]$. Lorsque k=0, x=s et $g(s)=0 \geq v(s,s)$. Supposons la propriété vraie pour $k \geq 0$, et considérons $x \in$ Atteints $_i$ tel que longueur($A_i[s,x]$)=k+1.

Posons $z=pred_i(x)$, nécessairement $z \in$ Atteints $_i$, par hypothèse d'induction, $g_i(z) \geq v(A_i[s,z])$, d'après l'invariant G7, $g_i(x) \geq g_i(z) + v(z,x)$, d'où l'on déduit $g_i(x) \geq v(A_i[s,z]) + v(z,x) = v(A_i[s,x])$.

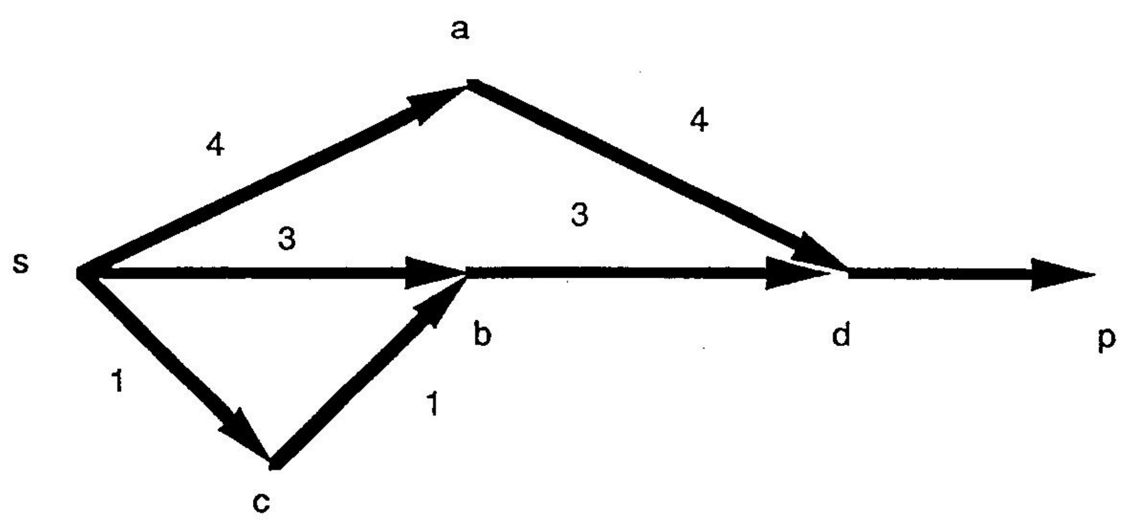
#

On peut en fait préciser légèrement l'invariant G9, à la manière de l'invariant G6.

Invariant G10: $\forall$ i, $1 \leq i \leq I$, $\forall$ $x \in$ Atteints $_i$, $g_i(x) = v(A_i[s,x])$ si $A_i[s,x]$ n'est constitué que de sommets de Fermes $_i$ (sauf éventuellement x lui-même) et $g_i(x) > v(A_i[s,x])$ sinon.

Preuve : immédiat, avec les invariants précédents.

#



	h1	h2
a	0	0
b	16	2
c	18	5
d	0	10

Avec la fonction heuristique h1, A* va se terminer (choisir le sommet p), en ayant trouvé le chemin [s,a,d,p] de valuation 18. Ainsi sur cet exemple, A* ne calcule pas le chemin optimal qui est [s,c,b,d,p] de valuation 15. Par contre si l'on utilise la fonction heuristique h2, qui est minorante, A* trouve bien ce chemin optimal.

On remarquera en outre qu'après avoir exploré s, a, b et c, nous avons $g_4(d)=6$. Et cette valeur ne correspond pas à la valuation du chemin de s à d dans l'arborescence A_4. De plus $g_4(d)$ est ne correspond pas à la valuation d'un chemin de s à d, constitué uniquement de fermés (sauf d).

Figure 2

Si les invariants précédents étaient implicites dans de nombreuses publications sur A*, à notre connaissance, le résultat suivant est original.

Théorème 1 : $\forall$ i, $1 \leq i \leq I$, le chemin A_i[s,Choix $_i$] n'est constitué que de sommets de Fermes $_i$ (sauf Choix $_i$ lui-même).

Nous reportons la preuve un peu technique de cet important résultat en annexe.

Corollaire: $\forall$ i, $1 \leq i \leq I$, posons x=Choix $_i$, $g_i(x)=v(A_i[s,x])= \min_\mu(v(\mu))$ où μ est un chemin de s à x constitué uniquement de Fermés i (sauf x).

Preuve: La première égalité vient du théorème 1 et de l'invariant G10. Quant à la deuxième elle découle du théorème 1 et de l'invariant G5.

#

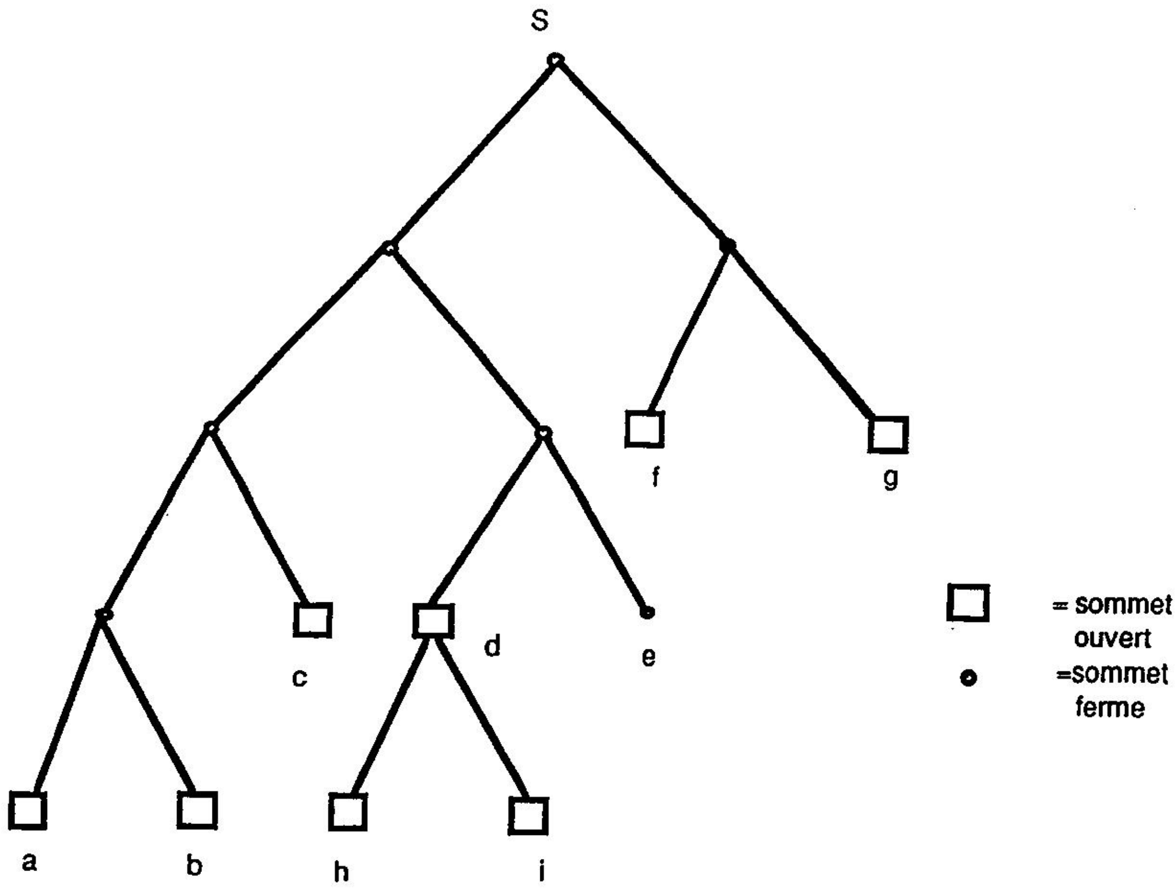
Ce corollaire permet de faire le lien avec l'algorithme de Dijkstra (1959), dont l'un des invariants principaux (cf. l'invariant C1 du III) est une généralisation simple de ce corollaire.

Une variante de A* plus efficace?

En effet ce théorème indique qu'à chaque étape seuls les ouverts x tels que A_i[s,x] n'est constitué que de sommets de Fermes $_i$ (sauf x lui-même), sont susceptibles d'être fermés (i.e. choisis par la fonction Choix).

Ainsi ce résultat engendre immédiatement une nouvelle implémentation de A*, en partitionnant la gestion de l'ensemble des ouverts en deux. D'un côté ceux qui sont susceptibles d'être choisis et de l'autre côté le reste des ouverts. Bien entendu ceci va engendrer une gestion plus délicate des ouverts, mais l'ensemble de ceux sur lesquels on sélectionne le nouveau sommet à explorer devrait être considérablement réduit (toutefois une réelle expérimentation de cette variante reste à faire).

Ce nouveau résultat permet d'appréhender l'algorithme A* comme un algorithme qui construit une arborescence de racine s, la partie utile de A_i (dont tous les noeuds internes sont fermés, et seules les feuilles sont ouvertes, cf. figure 3), en développant à chaque étape la feuille pour laquelle la valeur de f est minimale. Ceci permet de faire le lien avec les procédures dites "arborescentes" ou "séparation et évaluation" ("Branch and Bound").

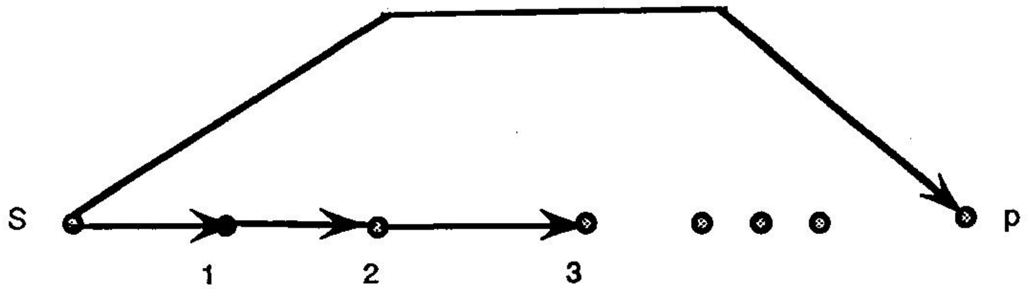


D'après le théorème 1, les seuls sommets ouverts que A* peut explorer à l'étape i, sur cette arborescence Ai sont: a, b, c, d, f, g. En effet h et i sont masqués par d. Enfin on remarquera sur cet exmple le sommet e qui est une feuille de l'arborescence qui n'admet pas de successeur.

Figure 3

Terminaison

Sans hypothèses plus précises que celles déjà formulées sur le graphe G, rien n'assure la terminaison en un nombre fini d'étapes de A*, comme l'indique l'exemple de la figure 4.



Cet exemple est constitué d'un chemin infini de s à p, dont les sommets sont indexés par les entiers naturels et d'un arc de s vers p.

$v(n,n+1)= 1/n(n+1)$, $v(s,p)=2$ et $h(n)=1/n$.

On montre par récurrence que sur cet exemple que A* ouvre successivement les sommets 1, 2, ..., n, ... avec: $g(n)=1-1/n$ et $f(n)=g(n)+h(n)=1$. Ainsi l'arc (s,p) n'est jamais utilisé pour atteindre p, et donc l'algorithme A* ne se termine pas en un temps fini.

Figure 4

Commençons par une propriété générale de terminaison:

Théorème 2: S'il existe $Y \subseteq X$, Y fini et tel que $\forall i$, $1 \leq i \leq I$, $\text{Choix}_i \in Y$, alors l'algorithme A* se termine en un temps fini.

Preuve: D'après l'invariant G3, pour chaque sommet $x \in \text{Atteints}_i$, $g_i(x)$ correspond à la valuation d'un chemin élémentaire de G. D'après les hypothèses ci-dessus, si $x=\text{Choix}\ i$, $g_i(x)$ correspond à la valuation d'un chemin de s à x dans le sous-graphe G(Y). Comme Y est fini, le nombre de chemins élémentaires est donc fini, donc aussi le nombre de valeurs différentes que peut prendre $g_i(x)$ lorsque x est choisi. On en déduit que le nombre d'étapes de l'algorithme est fini. Comme G est supposé localement fini, chaque étape est finie, et on peut ainsi conclure que l'algorithme se termine en un temps fini.

#

Ainsi nous pouvons conclure sous l'hypothèse forte de finitude du graphe G.

Proposition 1: Si G est fini, alors l'algorithme A*, se termine en un temps fini.

Preuve: Il suffit d'appliquer le résultat précédent avec Y=X.
#

Admissibilité

On supposera dorénavant, sauf mention contraire, que $f = g + h$.

Définition: On dit que A* est ***admissible***, lorsque A* trouve un chemin de valuation minimale lorsqu'il en existe un. Nous supposerons en outre que le chemin obtenu à partir de l'arborescence calculée par l'algorithme (A_I), a bien cette valuation minimale.

Il n'existe pas de condition nécessaire et suffisante connue d'admissibilité. Cependant il existe quelques conditions suffisantes. En particulier lorsqu'on impose des contraintes sur la fonction h.

Dans la suite nous supposerons que la fonction h vérifie la condition suivante: $\forall\ x \in X,\ h(x) \leq d(x,P)$. On dit alors que h est ***minorante***.
Une telle propriété se rencontre habituellement dans réseaux de connexion entre villles en prenant pour h la distance à vol d'oiseau. De même dans les problèmes de cheminement issus du VLSI, la distance de Manhattan est très utilisées comme fonction heuristique minorante.

Cette propriété intéressante nous permet d'obtenir un nouvel invariant de A*.

Invariant A1: Si x est choisi à l'étape i, alors $f_i(x) \leq d(s,P)$.

Preuve: Pour i=1, s est choisi à l'étape 1 et l'on a bien: $f_1(s)=h(s) \leq d(s,P)$, en utilisant la condition sur h.
Pour i>1, conformément aux hypothèses sur le graphe G, il existe un chemin μ de s à $p \in P$, de valuation d(s,P)[1]. Soit y le premier sommet à partir de s, de μ tel que $y \notin$ Fermés $_i$. Nécessairement y est adjacent à

[1] Lorsque d(s,P) n'est pas atteinte par un chemin, mais est la borne inférieure des valuations d'un ensemble de chemins, on peut se ramener à la démonstration ci-dessus.

un sommet de Fermés $_i$, car $i>1$, et donc $y \in$ Ouverts $_i$. Le choix de x comme sommet à explorer à l'étape i, implique:
$f_i(x)=g_i(x) + h(x) \leq g_i(y) + h(y) \leq g_i(y) + d(y,P)$.

D'après l'invariant G5, $g_i(y) \leq v(\mu[s, y])$. De plus $d(y,P) = v(\mu[y, p])$.
D'où, $f_i(x) \leq v(\mu[s, y]) + v(\mu[y, p]) = d(s,P)$.
Ainsi $f_i(x) \leq d(s,P)$.

#

Définition: soit $V=\{x\in X$ / il existe un chemin élémentaire μ de s à x tel que $\forall y \in \mu, v(\mu[s,y])+h(y) \leq d(s,P) \}$

Invariant A2: Si x est choisi à l'étape i, alors $x \in V$.

Preuve: d'après l'invariant G3, il existe un chemin élémentaire μ de s à x, tel que $g_i(x)=v(\mu)=v(\mu[s,x])$ et $\forall y \in \mu, y\neq x$, il existe $j < i$ tel que y=Choix $_j$ et $g_j(y)=v(\mu[s,y])$. En utilisant l'invariant A1, nous avons $g_i(x)+h(x) \leq d(s,P)$, et $\forall y \in \mu$, $g_j(y)+h(y) \leq d(s,P)$. On en déduit $x \in V$.

#

Proposition 2: Si V est fini alors A*, se termine en un nombre fini d'étapes.

Preuve: l'invariant A2 nous permet d'appliquer le théorème 2 en prenant Y=V.

#

Remarque: Certains auteurs considèrent des ε-graphes, i.e. des graphes vérifiant : $\forall xy \in U, v(xy) > \varepsilon > 0$. Il est facile de vérifier sur ces graphes que si $\forall x \in X, h(x)\geq 0$ et $d(s,P)<+\infty$, V est nécessairement fini. (En effet les sommets de V sont au plus à une distance en nombre d'arcs de $d(s,P)/\varepsilon$ du sommet s).

Dans la suite nous supposerons V fini. (En effet sur l'exemple de la figure 4, où V est infini, l'algorithme A*, muni d'une heuristique minorante, ne se termine pas en un temps fini).

En outre nous prendrons la convention suivante: $\forall p\in P, h(p)=0$. Il est à noter que cette condition reste compatible avec l'hypothèse "h minorante": $\forall x \in X, h(x)\leq d(x,P)$, car nous avions initialement pris l'hypothèse (ii) $\forall p',p'' \in P, d(p',p'')\geq 0$.

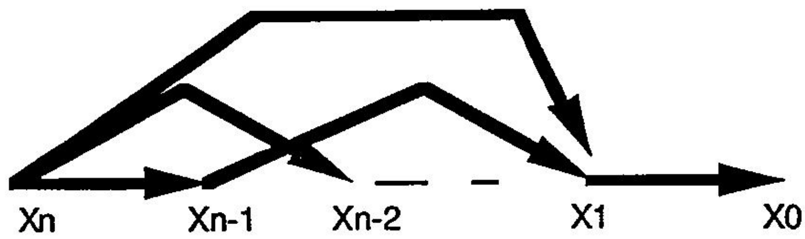
On retrouve ainsi un théorème énoncé dès les premiers travaux sur A*, Hart, Nilsson et Raphael (1972).

Théorème 3: S'il existe un chemin de s à P, alors l'algorithme A* s'arrête à l'étape I, sur un sommet $p \in P$, et l'on a $g_I(p)=v(A_I[s,p])=d(s,P)$.

Preuve: Supposons qu'il existe un chemin μ de s à $p \in P$, et que l'algorithme A* se termine avec un ensemble vide de sommets ouverts, sans avoir fermé de sommet de P.
Remarquons que le sommet s reste fermé au cours de l'algorithme (car il n'existe pas de circuit à valuation négative dans le graphe), et que les voisins des sommets fermés, non eux-mêmes fermés sont ouverts. Le sommet p étant non fermé avant l'étape I, il existe nécessairement sur le chemin μ un sommet ouvert, d'où la contradiction.
Pour conclure, il suffit d'utiliser l'invariant A1 à l'étape I, lorsque $p=\text{Choix}_I$ et $p \in P$. En effet $f_I(p)=g_I(p)+h(p)=g_I(p) \leq d(s,P)$ (Inv A1 et $h(p)=0$).
De plus d'après l'invariant G9, $v(A_I[s,p]) \leq g_I(p)$, et l'on a donc:
$d(s,P) \leq v(A_I[s,p]) \leq g_I(p) \leq d(s,P)$, d'où $g_I(p)=v(A_I[s,p])=d(s,P)$.

#

Ainsi dans ce cas A* est admissible, cependant comme le montre l'exemple classique de la figure 5, dû à Martelli, A* peut avoir un comportement exponentiel (i.e. faire un nombre exponentiel d'explorations de sommets).



Sur cet exemple d'un graphe à n+1 sommets, le sous-graphe engendré par Xn, X1 est un ordre total (avec tous les arcs y compris ceux de transitivité). Le sommet X0 est uniquement successeur de X1. En prenant la valuation et fonction heuristique définies, ci-dessous on peut vérifier que l'algorithme A* appliqué à la recherche d'un plus court chemin de Xn à X0, sur cet exemple va explorer un nombre exponentiel de sommets (de l'ordre de 2^{n-1}).

$v(X1,X0)=2^{2n}$,
$v(Xi,Xj)=2^{i-2}-2^{j-1}+i-j$ pour $1 \leq j < i \leq n$.
$h(Xi)=2^{i-1}+2i-3$.

De plus on peut remarquer que sur cet exemple, la fonction h proposée est minorante.

Figure 5

III Applications

a) Consistance

Définition: On dit que A* est ***consistante***, lorsque la fonction h vérifie l'une des deux conditions équivalentes suivantes:
(1) $\forall$ x,y $\in$ X, $h(x)-h(y) \leq d(x,y)$
(2) $\forall$ xy $\in$ U, $h(x)-h(y) \leq v(x,y)$

Dans ce cas on peut définir la valuation v' :
$\forall$ xy $\in$ U, $v'(xy)= v(xy)-h(x)+h(y)$. D'après (2) v' est positive.

Nous allons voir que l'algorithme de Dijkstra (1959) sur (G, v') se comporte comme A* sur (G, v,h), et l'on peut donc dans ce cas simplifier A*.
Rappelons que l'algorithme de Dijkstra peut se voir comme un cas particulier de A*, pour lequel la fonction heuristique h est nulle, et de plus le cas 2 de l'algorithme est supprimé, parce qu'inutile (en effet ce cas ne se produit jamais). Cependant il faut ajouter que l'algorithme de Dijkstra n'est valide que lorsque la valuation du graphe est positive ou nulle.

Proposition 3: Les algorihmes Dijkstra sur G muni de v' et A* sur G muni de v et h ont des comportements identiques (i.e. à chaque étape les ensembles de sommets fermés et ouverts sont identiques et le sommet choisi est le même).

preuve : Pour ce faire, nous allons d'abord montrer que l'algorithme A*, lorsque h est consistante ne réouvre jamais un sommet. Il suffit de prouver l'invariant:

Invariant C1: Si x est choisi à l'étape i, alors $g_i(x) = d(s,x)$.

Preuve de l'invariant: On a donc $d(s,x) < +\infty$.
Pour i=1, s est choisi à l'étape 1 on a bien $g(s)=0 \leq d(s,s)$.

Pour $i>1$, supposons que $d(s,x)$ soit atteinte, il existe un chemin μ de s à x, de valuation $d(s,x)$ (même remarque que dans la preuve de l'invariant A1). Soit y le premier sommet à partir de s, de μ tel que $y \notin$ Fermés $_i$. Nécessairement y est adjacent à un sommet de Fermés $_i$, car $i>1$, et donc $y \in$ Ouverts $_i$. Le choix de x comme sommet à explorer à l'étape i, implique:
$f_i(x)=g_i(x) + h(x) \leq g_i(y) + h(y)$.

D'après l'invariant G5, $g_i(y) \leq v(\mu[s, y]) = d(s,y)$.
D'où $g_i(x) \leq d(s,y) + h(y) - h(x) \leq d(s,y) + d(y,x)$. En utilisant la condition (1) de consistance.
De plus comme y appartient à μ qui est par hypothèse un chemin de valuation minimale joignant s à x, nous avons:
$d(s,y) + d(y,x) = d(s,x)$. Et donc $g_i(x) \leq d(s,x)$, ce qui permet de conclure en utilisant l'invariant G3.1.

#

Ainsi lorsque h est consistante on peut supprimer de l'algorithme A*, le cas 2. Pour conclure il suffit de remarquer qu'en faisant les changements:
$g'(x)=g(x)+h(x)-h(s)$
$v'(xy)= v(xy)-h(x)+h(y)$

Il y a identité parfaite entre les deux algorithmes:
- l'instruction $g'(s) \leftarrow 0$; est équivalente à $g(s) \leftarrow 0$;
- le choix du sommet ayant le plus petit valeur pour g' correspond au sommet minimisant $f=g+h$.
- le test $g'(y) > g'(x) + v'(xy)$ correspond à $g(y)> g(x) + v(xy)$.
- l'instruction $g'(y) \leftarrow g'(x) + v'(xy)$ correspond à $g(y) \leftarrow g(x) + v(xy)$.

D'où si l'on exécute A* avec v et h, et Dijkstra avec v', à chaque étape, les ensembles de sommets ouverts et fermés sont identiques, et le sommet choisi est le même.

#

Analogie entre A1 et C1

Lorsqu'on considère les preuves de ces deux invariants, il y a sans aucun doute une ressemblance. Ceci peut s'exprimer à l'aide de la généralisation commune suivante:

Proposition 4: Soit $Y \subseteq X$, tel que $\forall x \in X, \forall y \in Y, h(x)-h(y) \leq d(x,Y)$
si x est choisi à l'étape i, alors $\forall y \in Y, g_i(x)+h(x) \leq d(s,Y)+h(y)$.

Preuve: Pour $i=1$, s est choisi à l'étape 1 on a bien $g(s)+h(s)=h(s)\leq d(s,Y)+h(y)$, en utilisant l'hypothèse sur Y.
Pour $i>1$, supposons que $d(s,Y)$ soit atteinte, il existe un chemin μ de s à Y, de valuation $d(s,Y)$ (même remarque que dans la preuve de l'invariant A1). Soit z le premier sommet de μ à partir de s tel que $z \notin$ Fermés $_i$. Nécessairement z est adjacent à un sommet de Fermés $_i$, car $i>1$, et donc $z \in$ Ouverts $_i$. Le choix de x comme sommet à explorer à l'étape i, implique:
$f_i(x)=g_i(x) + h(x) \leq g_i(z) + h(z)$.

D'après l'invariant G5, $g_i(z) \leq v(\mu[s,z])$.
D'où $g_i(x)+h(x)\leq v(\mu[s,z])+h(z)\leq v(\mu[s,z])+h(y)+d(z,Y)$ pour tout $y\in Y$ (en utilisant l'hypothèse sur Y.
De plus comme y appartient à μ qui est par hypothèse un chemin de valuation minimale joignant s à Y, nous avons:
$v(\mu[s,z])+d(z,Y)= d(s,Y)$. Et donc $g_i(x)+h(x)\leq d(s,Y)+h(y)$, d'où le résultat.

#

En posant Y=P on obtient l'invariant A1, tandis qu'en choisissant Y={x} on retrouve C1. Ainsi l'hypothèse de consistance permet d'assurer une uniformité dans le comportement de l'algorithme A*. En effet ce qui dans le cas général n'est vrai que par rapport aux sommets de P, devient vrai pour tous les sommets du graphe. Ainsi peut-être faudrait-il dans ce cas remplacer l'adjectif "consistant" par "uniforme".

b) Comparaisons de plusieurs heuristiques A* sur un même graphe

A chaque nouvelle définition d'une fonction h, correspond un nouvel algorithme noté A*(h), le comportement de cet algorithme est modifié par l'action de la fonction Choix, les sommets ouverts et fermés vont donc varier suivant les fonctions h, appliquées sur un même graphe initial. Ainsi par exemple on peut envisager de comparer sur un même graphe, deux algorithmes $A^*(h_1)$ et $A^*(h_2)$ respectivement munis de deux fonctions heuristiques h_1 et h_2. On les supposera toutes deux minorantes. Lorsque $h_1(x)<h_2(x)$ il est possible de montrer que l'algorithme $A^*(h_2)$ explore moins de sommets que $A^*(h_1)$. Cependant ce résultat est dur à interpréter car $A^*(h_2)$ peut explorer plusieurs fois un même sommet. Ces notions ont été étudiées d'abord par Hart et al

(1968), puis Pohl (1970) et l'on trouvera une excellente synthèse dans Gelperin (1977).

Remarquons cependant que si une fonction h évolue au cours des itérations de l'algorithme, les résultats précédents restent valides à condition que la fonction h reste minorante. (Il faut toutefois remplacer dans la définition de l'ensemble V, h(y) par min $h_i(y)$).

On peut ainsi imaginer une fonction vérifiant $\forall x \in X$, $h_i(x) \leq d(x,P)$ (où i représente la ième étape de l'algorithme). On espère intuitivement obtenir une meilleure efficacité de l'algorithme lorsque $\forall x \in X$, $h_i(x) \leq h_{i+1}(x) \leq d(x,P)$ (i.e. l'estimation heuristique se rapproche de la valeur exacte). Remarquons toutefois que la complexité de l'implémentation peut augmenter, car la recherche du sommet ouvert minimisant f=g+h, va être plus coûteuse.

c) Variantes de A*

La preuve du théorème 3, nous permet d'exprimer une condition un peu plus générale d'admissibilité, en considérant la classe d'algorithmes A*, pour lesquels la fonction de Choix(A, f, g, h, P) renvoie le sommet de A ayant la plus petite valeur pour f (on ne suppose plus nécessairement que f=g+h).

Plaçons nous dans le cas où il existe un chemin de s à P et où l'algorithme A* se termine à l'étape I par le choix d'un sommet $p \in P$. On peut alors définir la condition suivante sur f:

CS1: pour tout chemin μ de valuation minimale de s à P, si y est le le premier sommet de μ non fermé à l'étape I, alors:

$f_I(p) \leq f_I(y) \Rightarrow g_I(p) \leq d(s,P)$

Proposition 5: si f vérifie CS1 alors A* est admissible.

preuve: immédiat, en utilisant un raisonnement analogue à celui utilisé au théorème 3. En effet plaçons nous à l'étape finale I lorsque p=Choix $_I$ et $p \in P$. Soit μ un chemin de valuation minimale de s à P et y le premier sommet de μ non fermé à l'étape I. Nécessairement $y \in$ Ouverts $_I$, et on a $f_I(p) \leq f_I(y)$. D'après la condition CS1, on en déduit $g_I(p) \leq d(s,P)$.
De plus d'après l'invariant G9, $v(A_I[s,p]) \leq g_I(p)$, et l'on a donc:
$d(s,P) \leq v(A_I[s,p]) \leq g_I(p) \leq d(s,P)$, d'où $g_I(p) = v(A_I[s,p]) = d(s,P)$ et donc A* est admissible.

#

Il est immédiat de vérifier que f=g+h vérifie bien la condition CS1. En outre si une fonction f vérifie CS1, alors pour toute fonction strictement croissante ϕ, $\phi$of vérifie CS1. Ainsi la fonction k(g+h) avec k réel positif.

Il est facile de montrer que des fonctions du type max(g,h) ou (g^2+h^2), vérifient aussi CS1, lorsque v et h sont des fonctions positives ou nulles.

Sur des graphes munis d'une valuation positive (et donc d'une fonction h aussi positive), Martelli (1977) a proposé une variante de A*, notée B, que l'on peut comprendre comme une modification de la fonction de choix. En effet la nouvelle fonction choix peut se définir par l'algorithme suivant (Figure 6):

```
Choix(A,F,f,g,h)

début

Pour tout sommet x de A
  faire    si g(x)+h(x)<F alors f(x) ← g(x);
           sinon f(x) ← g(x)+h(x);

a ← min (A,f);
{le sommet a atteint donc la valeur minimale de f sur A}
F ← max (F,g(a)+h(a));
Choix(A,F,f,g,h) ← a;

fin
```

Figure 6

L'instruction F ← 0; est ajoutée aux initialisations de A*. On obtient alors un algorithme intermédiaire "seuillé" entre Dijkstra et A*. Comme A* et B ne diffèrent que par la fonction de choix, les principaux invariants de A*, sont encore valides pour B.

Afin de montrer l'admissibilité de B, on supposera V fini, h minorante et qu'il existe un chemin de s à P.

Proposition 6: B est admissible.

preuve: Dans cette preuve, nous supposerons le graphe G fini. Le résultat reste vrai lorsque V est fini, mais la preuve de la terminaison dans ce cas est un peu plus compliquée (lorsque f≠g+h). Lorsque G est fini, il est trivial de vérifier que B se termine en un nombre fini d'étapes et Il suffit de montrer que B satisfait la condition générale CS1.

Plaçons nous dans les hypothèses de la condition CS1. Supposons $f_I(p) \leq f_I(y)$, nécessairement $f_I(p)=g_I(p)$ car $h(p)=0$ par hypothèse. Deux cas sont à considérer:
1°) $f_I(y)=g_I(y)$ mais alors d'après l'invariant G5, $g_i(y) \leq v(\mu[s, y]) = d(s,y)$ et comme de plus $v(\mu[y, P]) \geq 0$, on a bien: $g_i(p) \leq d(s,P)$.
2°) $f_I(y)=g_I(y)+h(y)$, la preuve est identique en utilisant que h est minorante.

#

Martelli (1977) a démontré sur B, deux autres résultats importants, dont les preuves un peu techniques sont omises ici :

- B admet un nombre d'étapes borné par $|V|^2$.
- A la condition que A* et B départagent les ex-aequos de la même manière, alors le nombre d'étapes de B est inférieur ou égal à celui de A*.

IV Annexe:

Théorème 1 : $\forall$ i, $1 \leq i \leq I$, le chemin $A_i[s, Choix_i]$ n'est constitué que de sommets de $Fermes_i$ (sauf $Choix_i$ lui-même).

Preuve: Pour ce faire nous allons utiliser une double induction (à la manière de la preuve de l'invariant G8 dont nous conservons exactement les notations). Ainsi nous allons prouver que la propriété Q_i suivante est un invariant de A*.
Soit K_i le nombre d'opérations pred(y) $\leftarrow x_0$, réalisées au cours de l'exploration de x_0. Notons $A_{i,j}$, $0 \leq j \leq K_i$, le graphe A_i après la jème opération. Ainsi $A_{i,0}= A_i, \dots, A_{i,K_i} = A_{i+1}$. Nous avons déjà montré que $A_{i,j}$ est une arborescence de racine s (cf. preuve de l'invariant G8).

Montrons par induction sur i et j la propriété suivante:
$Q_{i,j}$: a) $\forall x \in Atteints_{i,j}$, la fonction $\alpha \rightarrow f_{i,j}(y_\alpha)$ est strictement croissante (où $\{y_\alpha\}$ est la suite des ouverts à l'étape i,j du chemin $A_{i,j}[s,x]$).
b) $\forall z \in Fermés_{i,j} - A_{i,j}[s, Choix_i]$,
$\forall y \in Ouverts_{i,j} \cap DescA_{i,j}(z)$ on a $f_{i,j}(z) \leq f_{i,j}(y)$.

c) $A_{i,j}[s,\text{Choix}_i]-\{\text{Choix}_i\} \subseteq \text{Fermés}_{i,j}$

On note Q_i la propriété $Q_{i,0}$. Q_0 est trivialement vraie.On suppose Q_i vraie et nous allons montrer Q_{i+1}. On procède par induction sur j, pour j=0, comme $Q_{i,0}=Q_i$, la propriété $Q_{i,0}$ est vraie grâce à l'hypothèse d'induction sur i. Supposons $Q_{i,j}$ vraie, pour $0 \leq i < I$ et $0 \leq j < K_i$, et montrons qu'après une j+1ème modification du type $pred(y) \leftarrow x_0$, $Q_{i,j+1}$ est vraie.

Commençons par établir c). Il est clair que $y \notin A_{i,j}[s,\text{Choix}_i]$, car sinon $A_{i,j+1}$ ne serait pas une arborescence, et par hypothèse de récurrence, nous avons $A_{i,j}[s,\text{Choix}_i] \subseteq \text{Fermés}_{i,j}$. Donc:

$A_{i,j+1}[s,\text{Choix}_i]=A_{i,j}[s,\text{Choix}_i] \subseteq \text{Fermés}_{i,j+1}$, d'où le résultat.

Il nous reste à établir les propriétés a) et b).

A) Si $y \notin \text{Atteints}_{i,j}$ lorsqu'on considère le sommet y, l'arc x_0y permet d'ajouter le sommet y à l'arborescence. La propriété a) est trivialement vraie par induction, lorsque $x \neq y$ et lorsque $x=y$, il suffit de remarquer que y est le seul ouvert du chemin $A_{i,j+1}[s,y]$, en utilisant la condition c).

La propriété b) est vraie car cette opération ne modifie pas de couples (z,y) vérifiant les conditions de la propriété b) et donc reste vraie par l'hypothèse de récurrence.

Donc $Q_{i,j+1}$ est vraie.

B) Si $y \in \text{Atteints}_{i,j}$. $A_{i,j+1}$ se déduit de $A_{i,j}$ en remplaçant l'arc $(pred_i(y), y)$ par (x_0, y). Etudions la validité de la propriété a) pour un sommet $x \in \text{Atteints}_{i,j}$:

B1) si $x \notin \text{Desc}_{A_{i,j}}(y)$, alors la propriété a) est trivialement vraie par hypothèse de récurrence, car le chemin $A_{i,j+1}[s,x]$ n'a pas été modifié.

B2) Dans le cas contraire, deux cas sont à considérer:

B2.1) $y \in \text{Ouverts}_{i,j}$. La propriété est vraie pour sur $A_{i,j+1}]y,x]=A_{i,j}]y,x]$, par hypothèse de récurrence. De plus y est le seul ouvert du chemin $A_{i,j+1}[s,y]$ et comme $f_{i,j+1}(y)<f_{i,j}(y)$, la propriété est donc vraie sur $A_{i,j+1}[s,x]$.

B2.2) $y \in \text{Fermes}_{i,j}$. Donc y est réouvert à l'étape i. La propriété est vraie pour $A_{i,j+1}]y,x]=A_{i,j}]y,x]$, par hypothèse de récurrence. De plus y est le seul ouvert du chemin $A_{i,j+1}[s,y]$ et comme $f_{i,j+1}(y)<f_{i,j}(y)$, en utilisant l'hypothèse d'induction sur b) qui relie $f_{i,j}(y)$ aux valeurs de f des sommets ouverts sur le chemin $A_{i,j}]y,x]$, la propriété est donc vraie sur $A_{i,j+1}[s,x]$.

La propriété b) se vérifie facilement à l'aide des hypothèses de récurrence, car dans ce cas l'opération sur l'arborescence ne fait que diminuer le nombre de couples (y,z) possibles pour cette propriété b) sans changer les valeurs de la fonction f associées aux couples restant.

Donc $Q_{i,j+1}$ est vraie.

Donc Q_{i,K_i} est vraie. Il nous reste à montrer $Q_{i+1,0}$. Cela veut dire qu'il faut montrer que les propriétés b) et c) restent vraies en changeant Choix_i, par Choix_{i+1}. C'est évident pour c), considérons b).

Soit (y,z) un tel couple $z \in \text{Fermés}_{i+1} - A_{i+1}[s, x_1]$ et $y \in \text{Ouverts}_{i+1} \cap \text{DescA}_{i+1}(z)$. Notons $x_1 = \text{Choix}_{i+1}$ et $D_j(z) = \text{DescA}_j(z)$ $\Delta_j(z) = \text{Atteint}_j - D_j(z)$. On remarque $z \notin A_{i+1}[s, x_1]$ et donc $x_1 \in \Delta_{i+1}(z)$ Comme $y \in \text{Ouverts}_{i+1} \cap D_{i+1}(z)$, d'après le lemme ci-dessous $f_{i+1}(z) \leq f_{i+1}(y)$, d'où le résultat.

#

Lemme: Notons $x_1 = \text{Choix}_{i+1}$ et considérons un sommet $z \in \text{Fermes}_{i+1}$, alors :

- si $x_1 \in D_{i+1}(z)$ alors $\forall\ y \in \text{Ouverts}_{i+1} \cap \Delta_{i+1}(z)$ on a $f_{i+1}(z) \leq f_{i+1}(y)$
- si $x_1 \in \Delta_{i+1}(z)$ alors $\forall\ y \in \text{Ouverts}_{i+1} \cap D_{i+1}(z)$ on a $f_{i+1}(z) \leq f_{i+1}(y)$

Preuve: on considère un tel sommet $z \in \text{Fermes}_{i+1}$ et l'on pose: $k = \max\{i' - 1 / i' < i+1 \text{ et } z \in \text{Ouverts}_{i'}\}$. Nécessairement $z = \text{Choix}_{k+2}$, et pour tout i', $k+1 \leq i' \leq i$, $f_{i'}(z) \leq f_{i+1}(z)$. Montrons par induction sur i', $k \leq i' \leq i+1$, la propriété suivante $R_{i'}(z)$.

$R_{i'}(z)$: si $x_1 = \text{Choix}_{i'+1}$

- si $x_1 \in D_{i'+1}(z)$ alors $\forall\ y \in \text{Ouverts}_{i'+1} \cap \Delta_{i'+1}(z)$ on a $f_{i'+1}(z) \leq f_{i'+1}(y)$
- si $x_1 \in \Delta_{i'+1}(z)$ alors $\forall\ y \in \text{Ouverts}_{i'+1} \cap D_{i'+1}(z)$ on a $f_{i'+1}(z) \leq f_{i'+1}(y)$

Lorsque i'=k, alors $x_1 = z \in D_{k+1}(z)$.

Comme $f_{k+1}(z) = \min_{y \in \text{Ouverts}_{k+1}} \{f_{k+1}(y)\}$, nécessairement $R_k(z)$ est vérifiée.

Supposons maintenant $R_{i'-1}(z)$, $k+1 \leq i' \leq i$, vraie et montrons $R_{i'}(z)$. Notons $t = \text{Choix}_{i'+1}$. Deux cas sont possibles:

1°) $t \in D_{i'}(z)$, alors par hypothèse de récurrence:
$\forall\ y \in \text{Ouverts}_{i'} \cap \Delta_{i'}(z)$ on a $f_{i'}(z) \leq f_{i'}(y)$.

Considérons maintenant un sommet $y \in Ouverts_{i'+1} \cap \Delta_{i'+1}(z)$, nécessairement $y \in Ouverts_{i'}$, sinon on devrait avoir $y \in D_{i'+1}(z)$. En outre $y \in \Delta_{i'}(z)$ (car $y \in \Delta_{i'+1}(z) \subseteq \Delta_{i'}(z)$). Donc $f_{i'}(z) \leq f_{i'}(y)$.

D'autre part $f_{i'}(y)=f_{i'+1}(y)$, sinon on aurait $y \in D_{i'+1}(z)$ comme successeur de t. Comme $f_{i'}(z)=f_{i'+1}(z)$, nous avons donc:
$\forall\, y \in Ouverts_{i'+1} \cap D_{i'+1}(z)$ on a $f_{i'+1}(z) \leq f_{i'+1}(y)$.

Lorsque $t \in D_{i'+1}(z)$, alors $R_{i'}(z)$ est vraie.

Lorsque $t \in \Delta_{i'+1}(z)$, alors $t \in Ouverts_{i'+1} \cap \Delta_{i'+1}(z)$ et donc $f_{i'+1}(z) \leq f_{i'+1}(t)$, car $f_{i'+1}(t)=\min_{y \in Ouverts_{i'+1}} \{f_{i'+1}(y)\}$.

Donc $\forall\, y \in Ouverts_{i'+1} \cap D_{i'+1}(z)$, $f_{i'+1}(z) \leq f_{i'+1}(t) \leq f_{i'+1}(y)$ et donc $f_{i'+1}(z) \leq f_{i'+1}(y)$ et ainsi $R_{i'}(z)$ est vraie.

2°) $t \in \Delta_{i'}(z)$, la démonstration est identique à celle du cas précédent à condition d'échanger les rôles de D et Δ.

#

Remarque: il est facile de vérifier dans les preuves ci-dessus, que la validité du théorème 1, n'impose aucune condition sur la fonction f, simplement que le choix du sommet à explorer se fait à chaque étape en prenant celui dont la valeur de f est minimale.

Remerciements: Que les nombreux collègues et étudiants de Montpellier, Lyon ou Brest qui ont subi avec sérénité les premières versions de ce travail soient ici remerciés.

V REFERENCES

R.K. Ahuja, K. Mehlhorn, J.B. Orlin, R.E. Tarjan, "Faster algorithms for the shortest path problem", JACM, vol 37, N°2 (1990) 213-233.

A. Bagghi, A. Mahanti, "Search algorithms under different kinds of heuristics - a comparative study", JACM, vol 13 , 3 (1983) 1-21.

H.J. Berliner, "The B* tree search algorithm: a best first proof procedure", Artificial Intelligence 10, 2 (1979) 201-214.

M. Bertier, M. Habib, "Contour extraction via path algorithms", (1983) 29-39,Trauner Verlag, Linz, Austria.

R. Dechter, J. Pearl, "The optimality of A*", in L. Kanal, V. Kumar Eds., "Search in Artificial Intelligence", Symbolic Computation Series, Springer-Verlag, 1988.

R. Dechter, J. Pearl, "Generalized best-first search strategies and the optimality of A*, JACM, vol. 32, 3 (1985) 505-536.

E.W. Dijkstra, "A note on two problems in connexion with graphs", Numerische Matematik, vol. 1 (1959) 269-271.

H. Farreny, M. Ghallab, "Eléments d'intelligence artificielle", Hermès, Paris, 1987.

M.L. Fredman et R.E. Tarjan, "Fibonacci heaps and their uses in improved network optimization algorithms", JACM, 1984,

D. Gelperin, "On the optimality of A*", Artificial Intelligence 8(1977) 69-76.

L.R. Harris, "The heuristic search under conditions of error", Artificial Intelligence 5(1974) 217-234.

P. Hart, N.J. Nilsson, B. Raphael, "A formal basis for the heuristic determination of minimum cost paths", IEEE Trans. Systems Man Cybernet. 4 (1968) 100-107.

P. Hart, N.J. Nilsson, B. Raphael, "Correction to a formal basis for the heuristic determination of minimum cost paths", Sigart Newsletter (1972) 28-29.

N. Huyn, R. Dechter, J. Pearl, "Probabilistic analysis of the complexity of A*", Artificial Intelligence 15 (1980) 241-254.

T. Ibaraki, "Theoretical comparisons of search strategies in branch and bound algorithms", Internat. J. of Computer and Information Sciences, 5,4 (1976) 315-344.

E.L. Lawler, M.G. Luby, B. Parker, "Finding shortest paths in very large networks", Proc. of WG'83, Workshop on Graphtheoretic Concepts in Computer Science, Eds. M. Nagl and J. Perl, Trauner Verlag (1983) 185-199.

A. Martelli, "Edge detection using heuristic search methods", Computer Graphics and Image processing 1 (1972) 169-182.