MODÉLISATION D'UN RÉSEAU ROUTIER À PARTIR D'UN GRAPHE AVEC DES ARCS DE LARGEUR, D'ORIENTATION ET DE PENTE ARBITRAIRES

Soutien au travail de groupe

Résumé

Conventions et terminologie
Modélisati
on Mouvement interactif (contrôlé par l'utilisateur)
Mouvement automatique (contrôlé par ordinateur)
Placement du personnage dans la scène

João Paulo Pereira
jjp@isep.ipp.pt

Index

Index des figures

Conventions et terminologie

Scène

- **Direction et sens du vecteur *haut*** : ceux correspondant à l'axe Z positif[1] .

Nœud n_i

- **Localisation** : point de coordonnées (x_i, y_i, z_i) ;
- **Largeur** : w_i (la largeur d'un nœud sera égale à la plus grande des largeurs des arcs convergeant/divergeant vers/depuis ce nœud).

Arc a_{ij}

- **Connexion** : du nœud n_i au nœud n_j ;
- **Inégalité** : $h_{ij} = z_j - z_i$;
- **Longueur** : s_{ij} ;
- **Largeur** : w_{ij} ;
- **Orientation** : α_{ij} = arctan[2] ((y_j - y_i) / (x_j - x_i)) (en radians) ;
- **Pente** : βij (en radians).

Caractère

- **Hauteur** : *HEIGHT_PERSONAGE* ;
- **Localisation** (centre géométrique) : point de coordonnées (x_P, y_P, z_P) ;
- **Orientation** : *dir* (en radians) ;
- **Vitesse horizontale** : *ancienne* ;
- **Vitesse verticale** (uniquement pour le mouvement automatique) : *velv* ;
- **Vitesse angulaire** (uniquement pour le mouvement automatique) : *voile*.

Animation (uniquement pour le mouvement automatique)

- **Circulation** : à droite ;
- **Nombre d'images qui constituent l'animation d'un mouvement élémentaire** : *n* ;
- **Temps écoulé entre les images** : supposé constant et égal à 1 ;
- **Rayons de courbure des mouvements élémentaires *B* et *F* :** *RAIOB* et *RAIOF* ;
- **Vitesses maximales pour les mouvements élémentaires *A* à *F* :** *VELA*, *VELB*, *VELC, VELD, VELE*, *VELF*.

---

[1] Notez que, par défaut, three.js suppose que le demi-axe Y positif correspond à la direction et au sens de ce vecteur [7].
[2] Vous devriez utiliser Math.atan2() au lieu de Math.atan() [5].

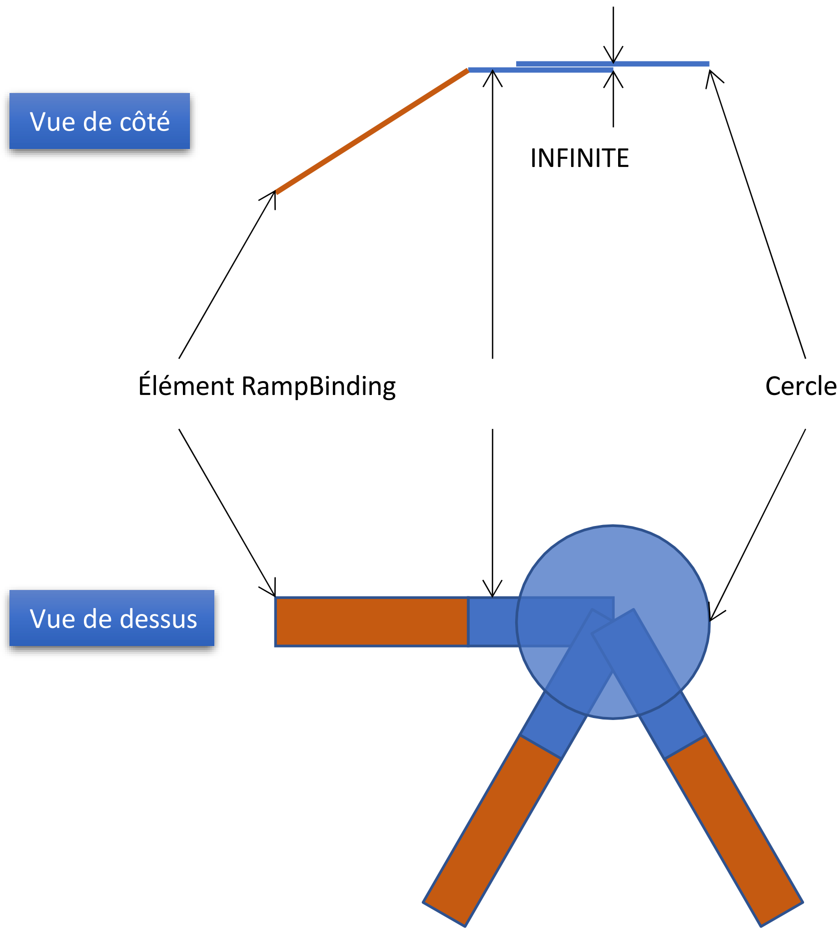

Figure 1 - Modèle du réseau routier

Modélisation

Le réseau routier peut être modélisé comme suit (figure 1).

Modélisation d'un nœud

La géométrie associée à un nœud n_i pourrait être celle d'un rond-point composé des éléments suivants :

- un cercle ;
- autant d'éléments de connexion qu'il y a d'arcs convergeant/divergeant sur/depuis ce nœud.

Cercle

Le cercle doit avoir les propriétés suivantes

- **centre** : (*xi*, *yi*, *zi* + *INFINITEMENT*) ;
- **rayon** : r_i = *K_CIRCLE* * *wi* / 2.0 ;

    où *K_CIRCLE* signifie une constante supérieure à 1,0 (par exemple, *K_CIRCLE* = 2,1).

Élément de connexion

Étant donné un nœud n_i connecté à un arc *aij*, la géométrie de l'élément de connexion peut être celle d'un rectangle horizontal ayant les propriétés suivantes :

- **longueur** : $s_i = K\_LINK * r_i$;
  où *K_LINK* est une constante supérieure à 1,0 (par exemple, *K_LINK* = 1,1) ;
- **largeur** : *wij* ;
- **orientation** : *αij*.

Modélisation d'un arc

La géométrie associée à un arc *aij* peut être celle d'une rampe (c'est-à-dire un rectangle incliné) ayant les propriétés suivantes :

- **longueur de la projection dans le plan OXY** : $p_{ij} = \sqrt{(xj - xi)^2 + (yj - yi)^2} - s_i - sj$;
- **gradient** : $hij = zj - zi$;
- **longueur** : $s_{ij} = \sqrt{(p_{ij2} + hij2)}$;
- **largeur** : *wij* ;
- **orientation** : *αij* ;
- **pente** : $\beta ij = \arctan(hij / p_{ij})$.

Mouvement interactif (contrôlé par l'utilisateur)

Il convient de conserver un enregistrement actualisé de l'emplacement du personnage dans le graphe, c'est-à-dire de savoir s'il se trouve à un nœud donné (le nœud $_{ni}$) ou sur un arc donné (l'arc *aij*).

S'il n'y avait pas de collision, l'emplacement du personnage dans la trame suivante serait donné par les équations suivantes (uniquement l'abscisse et l'ordonnée ; les coordonnées seront calculées plus tard)[3] :

- $x'_P = xP +_{ancien} * \cos(dir)$;
- $y'_P = yP +_{ancien} * \sin(dir)$.

Détection des collisions sur un nœud

Si le personnage se trouve au nœud $_{ni\ du}$ graphe, il n'y aura pas de collision si le point correspondant au nouvel emplacement lui appartient :

- au cercle de ce nœud ;
- à l'un des éléments de liaison de ce noeud ;
- à l'un des arcs convergeant/divergeant vers/depuis ce nœud.

Détection des collisions d'arc

Si le personnage se trouve sur l'arc *aij du* graphe, il n'y aura pas de collision si le point correspondant au nouvel emplacement lui appartient :

- à cet arc ;
- à l'élément de connexion de cet arc au nœud $_{ni}$;
- au cercle du nœud $_{ni}$;
- à l'élément de connexion de cet arc au nœud *nj* ;
- au cercle du noeud *nj*.

Détermination de l'appartenance

Affectation d'un point à un nœud

Le point correspondant au nouvel emplacement du caractère appartiendra au nœud $_{ni}$ si et seulement s'il appartient au cercle ou à l'un des éléments de connexion qui le représentent.

Appartenance d'un point à un cercle

Le point correspondant à la nouvelle localisation du personnage appartiendra au cercle du nœud $_{ni}$ si et seulement si sa distance par rapport au centre du cercle n'est pas supérieure au rayon :

- $(x'_P - xi)^2 + (y'_P - yi)^2 \leq r_i.^2$

En vérifiant cette condition, on considère que le personnage devient (s'il ne l'était pas déjà) situé au nœud $_{ni\ du}$ graphe. Les coordonnées du nouvel emplacement seront données par les équations :

- $xP = x'_P$;
- $yP = y'_P$;
- $zP = zi + HEIGHT\_PERSONAGE / 2.0$.

[3] On suppose, pour simplifier les calculs, que le temps entre les images est constant et égal à 1.

L'appartenance d'un point à un élément de liaison

Pour déterminer si le point correspondant au nouvel emplacement du caractère appartient à l'élément de liaison du nœud $_{ni}$ à l'arc *aij*, on peut procéder comme suit :

Un changement de système de coordonnées est effectué si les conditions suivantes sont remplies :

- faire correspondre la nouvelle origine avec le point (*xi*, *yi*) ;
- aligner le nouvel axe X sur l'axe longitudinal de l'élément de liaison.

Dans ce nouveau système, les coordonnées correspondant au nouvel emplacement du personnage seront données par les équations suivantes :

- $x''_P = (x'_P - xi) * \cos(\alpha ij) + (y'_P - yi) * \sin(\alpha ij)$;
- $y''_P = (y'_P - yi) * \cos(\alpha ij) - (x'_P - xi) * \sin(\alpha ij)$.

Le point correspondant au nouvel emplacement du caractère appartiendra à l'élément de liaison si et seulement s'il ne dépasse pas les limites du rectangle qui le représente :

- $0{,}0 \leq x''_{P \leq si}$;
- $-wij / 2.0 \leq y''_{P \leq} wij / 2.0$.

En vérifiant ces conditions, on considère que le personnage est maintenant (s'il ne l'était pas déjà) situé au nœud $_{ni}$ du graphe. Les coordonnées du nouveau lieu seront données par des équations identiques à celles de l'appartenance à un cercle :

- $xP = x'_P$;
- $yP = y'_P$;
- $zP = zi + HEIGHT\_PERSONAGE / 2.0$.

Appartenance d'un point à un arc

Pour déterminer si le point correspondant au nouvel emplacement du personnage appartient à l'arc *aij*, on peut procéder comme suit :

Le changement de système de coordonnées s'effectue de la même manière que pour déterminer si un élément de liaison en fait partie, c'est-à-dire si les conditions suivantes sont remplies :

- faire correspondre la nouvelle origine avec le point (*xi*, *yi*) ;
- aligner le nouvel axe X avec l'axe longitudinal de la projection de l'arc dans le plan OXY.

Dans ce nouveau système, les coordonnées correspondant au nouvel emplacement du personnage seront, comme précédemment, données par les équations suivantes :

- $x''_P = (x'_P - xi) * \cos(\alpha ij) + (y'_P - yi) * \sin(\alpha ij)$;
- $y''_P = (y'_P - yi) * \cos(\alpha ij) - (x'_P - xi) * \sin(\alpha ij)$.

Le point correspondant au nouvel emplacement du personnage appartiendra à l'arc si et seulement s'il ne dépasse pas les limites de la projection du rectangle qui le représente :

- $_{si} < x''_P < _{si} + _{pij}$;

- $-wij / 2.0 \leq y''_P \leq wij / 2.0$.

Si ces conditions sont vérifiées, le caractère est considéré comme étant (s'il ne l'était pas déjà) situé sur l'arc *aij du* graphe. Les coordonnées du nouvel emplacement seront données par des équations similaires à celles de l'appartenance à un cercle et à un élément de liaison. La seule différence réside dans l'inclusion d'une simple règle de trois dans le calcul de la coordonnée :

- $xP = x'_P$;
- $yP = y'_P$;
- $zP = zi + (x''_P - s_i) / p_{ij} * hij + HEIGHT\_PERSONAGE / 2.0$.

Mouvement automatique (contrôlé par ordinateur)

L'objectif est de déplacer le personnage d'un nœud d'origine à un nœud de destination, selon un itinéraire préalablement établi, qui est constitué d'une séquence de mouvements entre nœuds adjacents. Chacun de ces mouvements peut à son tour être décomposé en une séquence de six mouvements élémentaires : droit, pour se déplacer le long des rampes et d'une partie des carrefours giratoires ; et circulaire, pour entrer, circuler et sortir des carrefours giratoires.

Sur les rampes et les éléments de liaison, on suppose que le personnage se déplace sur la voie de circulation correspondant au côté droit de la chaussée. La distance entre le personnage et la bordure latérale droite peut être affectée de la valeur suivante :

- $_{bij}$ = *K_BERMA * wij* ;
  où *K_BERMA* désigne une constante telle que 0,0 < *K_BERMA* < 0,5 (par exemple, *K_BERMA* = 0.25).

Dans les cercles, on suppose que le personnage se déplace dans la direction directe[4] . La distance entre le caractère et la périphérie du cercle peut se voir attribuer la valeur suivante :

- $_{bi}$ = *K_BERMA * wi.*

En supposant que le personnage soit correctement situé et orienté, les mouvements élémentaires ci-dessus pourraient être les suivants (Figure 2) :

- **mouvement *A*** : circulaire direct (le personnage parcourt une partie du cercle) ;
- **mouvement *B*** : circulaire rétrograde[5] (le personnage sort du cercle) ;
- **mouvement *C*** : rectiligne (le personnage traverse une partie de l'élément de liaison) ;
- **mouvement *D*** : rectiligne (le personnage parcourt toute la longueur de la rampe) ;
- **mouvement *E*** : rectiligne (le personnage traverse une partie de l'élément de liaison) ;
- **mouvement *F*** : circulaire rétrograde (le personnage entre dans le cercle).

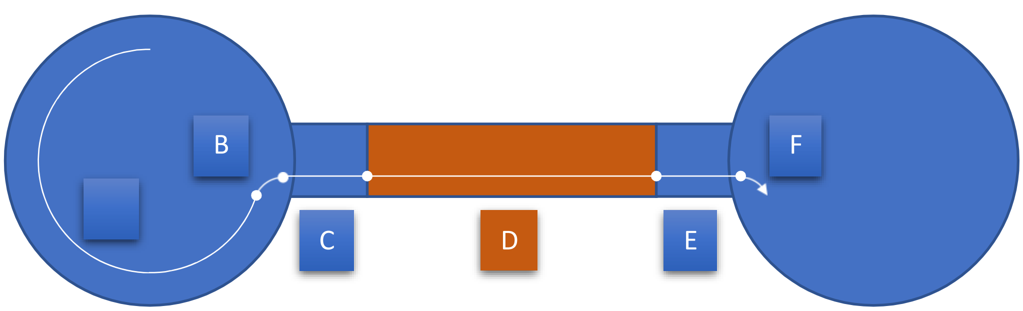

Figure 2 - Séquence des mouvements élémentaires

Le processus d'automatisation du mouvement comprendra les étapes suivantes :

[4] C'est-à-dire dans le sens inverse des aiguilles d'une montre.
[5] C'est-à-dire dans le sens du mouvement des aiguilles d'une montre.

1. initialisation de la position et de l'orientation du caractère ;
2. tant que le nœud de destination n'est pas atteint :
    a. préparation à l'animation du mouvement élémentaire *A* ;
    b. animation du mouvement élémentaire *A* ;
    c. préparation à l'animation du mouvement élémentaire *B* ;
    d. animation du mouvement élémentaire *B* ;
    e. préparation à l'animation du mouvement élémentaire *C* ;
    f. animation du mouvement élémentaire *C* ;
    g. préparation à l'animation du mouvement élémentaire *D* ;
    h. animation du mouvement élémentaire *D* ;
    i. préparation à l'animation du mouvement élémentaire *E* ;
    j. animation du mouvement élémentaire *E* ;
    k. préparation à l'animation du mouvement élémentaire *F* ;
    l. animation du mouvement élémentaire *F*.

L'étape *1* ne peut être réalisée qu'une seule fois, au début du processus. Les étapes *a* à *l* doivent être répétées autant de fois que nécessaire pour atteindre le nœud de destination.

Pour faciliter la compréhension, ce document commencera par décrire les étapes de préparation pour l'animation des mouvements de base *D*, *E, C, F,* B et *A,* dans cet ordre (étapes *g, i*, *e, k, c* et *a,* respectivement). Elle est suivie de la description de l'animation des mouvements élémentaires (communs aux étapes *h*, *j, f, l, d* et *b*). Enfin, l'initialisation de la position et de l'orientation du personnage est décrite (étape *1*).

Préparation à l'animation des mouvements de base

Pour préparer l'animation de chacun des mouvements élémentaires, il sera nécessaire de définir quatre paramètres :

- nombre d'images constituant l'animation du mouvement élémentaire : *n* ;
- vitesse angulaire du caractère : $_{voile}$;
- vitesse horizontale du personnage : $_{vieux}$;
- vitesse verticale du caractère : *velv*.

En outre, la nature discontinue des mouvements circulaires aura les implications suivantes :

- Dans chaque image, le personnage va parcourir, non pas un arc de cercle, mais la chaîne de caractères correspondante [1] (Figure 3) ;
- avant de commencer l'animation d'un mouvement circulaire, vous devez ajuster l'orientation du personnage, en soustrayant la moitié de la valeur de la vitesse angulaire : -vitesse / 2.0 ;
- une fois l'animation d'un mouvement circulaire terminée, vous devez réajuster l'orientation du personnage, en ajoutant la moitié de la valeur de la vitesse angulaire : $_{voile}$ / 2.0.

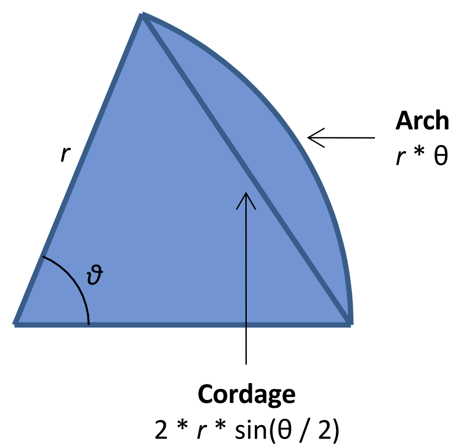

Figure 3 - Cordage

Mouvement *D*

Soit $_{ni\ le}$ nœud de provenance du personnage, *nj* le nœud adjacent où le personnage est dirigé, et *VELD* la vitesse maximale souhaitée pour ce mouvement.

Le personnage devra parcourir toute la longueur de la rampe.

Le nombre d'images qui composent l'animation sera donné par le plafond[6] du rapport entre la longueur de la rampe et cette vitesse :

- $n = \lceil sij / VELD \rceil$.

Étant donné la nature rectiligne du mouvement, la vitesse angulaire sera nulle :

- $_{voile} = 0{,}0$.

La vitesse horizontale sera donnée par le rapport entre la longueur de la projection de la rampe dans le plan OXY et le nombre d'images :

- $_{velh} = {}_{pij} / n$.

La vitesse verticale sera donnée par le rapport entre la pente de la rampe et le nombre d'images :

- $velv = hij / n$.

[6] La fonction Math.ceil() [6] peut être utilisée.

Mouvement *E*

Soit $_{ni}$ le nœud de provenance du personnage, *nj* le nœud adjacent dans lequel le personnage va entrer, *RAIOF* le rayon de courbure prévu pour le mouvement circulaire rétrograde *F* et *VELE* la vitesse maximale prévue pour le mouvement *E* (Figure 4).

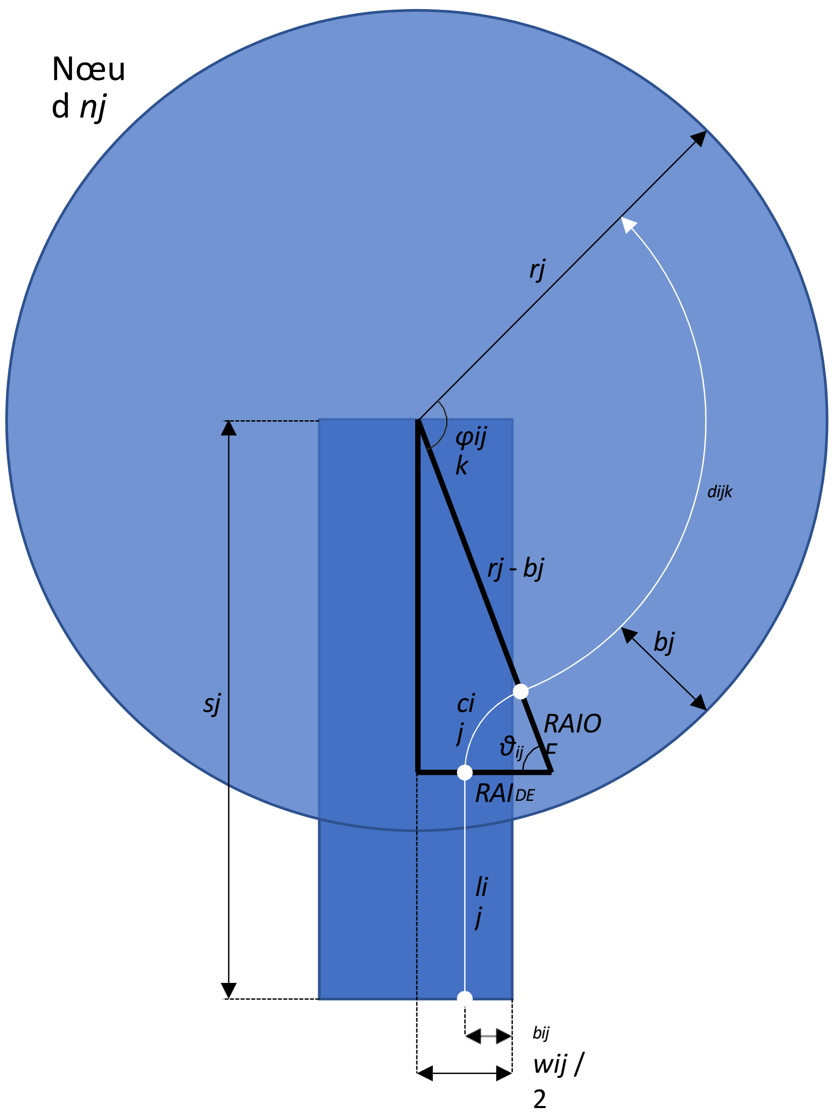

Figure 4 - Entrée d'un nœud

Le personnage ne devra parcourir qu'une partie de la longueur de l'élément de liaison, qui peut être déterminée à l'aide du triangle rectangle illustré sur la figure.

La longueur de l'hypoténuse du triangle rectangle sera donnée par l'équation :

- *hanche* = rj - *bj* + *RAIOF*.

La longueur de l'obliquité transversale sera donnée par l'équation :

- $cattrans = wij / 2.0 - _{bij} + RAIOF$.

En appliquant le théorème de Pythagore, on obtient pour la longueur de l'oblique longitudinale :

- $catlong = \sqrt{(hip^2 - cattrans2)}$.

La longueur de la course sera donnée par la longueur de l'élément de liaison soustraite de la longueur de l'obliquité longitudinale :

- $lij = sj - catlong$.

Le nombre d'images qui composent l'animation sera donné par le plafond du rapport entre la longueur du trajet et la vitesse :

- $n = \lceil lij / VELE \rceil$.

Étant donné la nature rectiligne du mouvement, la vitesse angulaire sera nulle :

- $_{voile} = 0,0$.

La vitesse horizontale sera donnée par le rapport entre la longueur de la course et le nombre d'images :

- $_{velh} = lij / n$.

Comme l'élément de liaison est horizontal, la vitesse verticale sera nulle :

- $velv = 0,0$.

Mouvement *C*

Soit *nj* le nœud duquel le personnage sortira, *nk* $_{le}$ nœud adjacent vers lequel le personnage se dirige, *RAIOB* le rayon de courbure prévu pour le mouvement circulaire rétrograde *B* et *VELC* la vitesse maximale prévue pour le mouvement *C* (figure 5).

En appliquant le même raisonnement que pour la préparation du mouvement *E*, le personnage n'aura à parcourir qu'une partie de la longueur de l'élément de liaison, qui peut être déterminée à l'aide du triangle rectangle représenté sur la figure.

La longueur de l'hypoténuse du triangle rectangle sera donnée par l'équation :

- $hip = rj - bj + RAIOB$.

La longueur de l'obliquité transversale sera donnée par l'équation :

- $cattrans = wjk / 2.0 - bjk + RAIOB$.

En appliquant le théorème de Pythagore, on obtient pour la longueur de l'oblique longitudinale :

- $catlong = \sqrt{(hip^2 - cattrans2)}$.

La longueur de la course sera donnée par la longueur de l'élément de liaison soustraite de la longueur de l'obliquité longitudinale :

- *ljk = sj - catlong*.

Le nombre d'images qui composent l'animation sera donné par le plafond du rapport entre la longueur du trajet et la vitesse :

- *n* = ⌈ljk / *VELC*⌉.

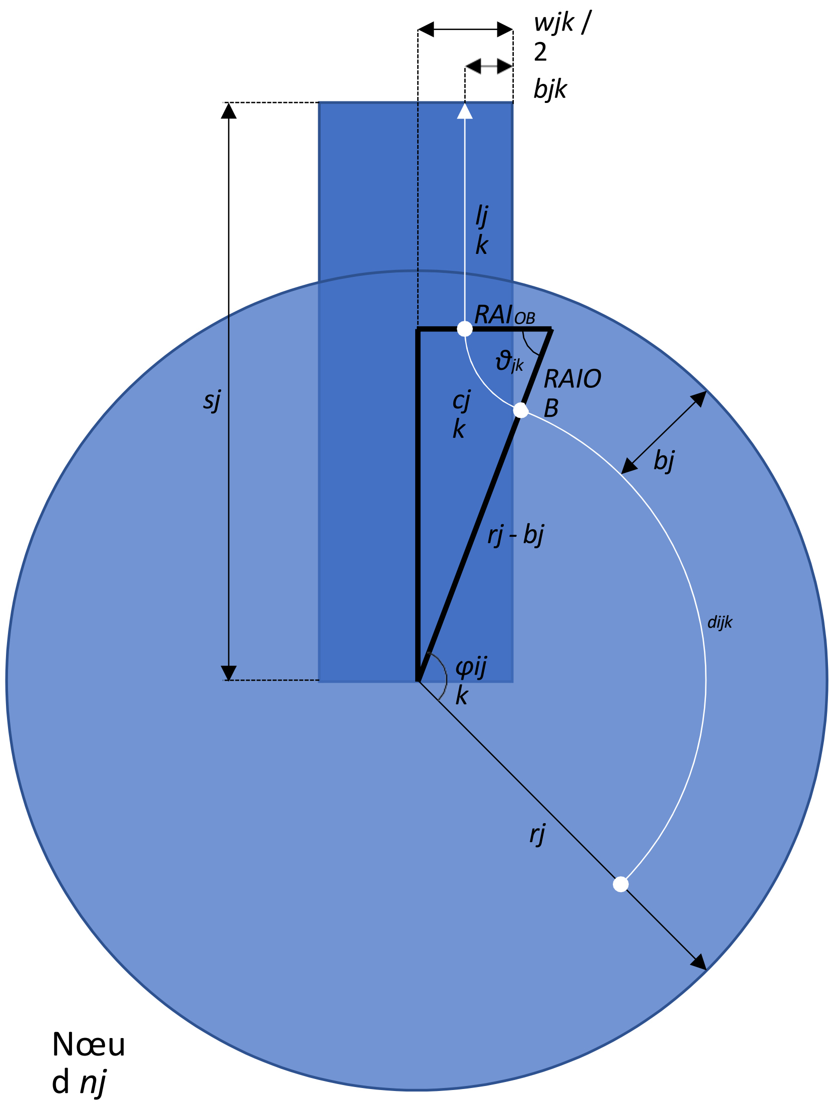

Figure 5 - Sortie d'un nœud

Étant donné la nature rectiligne du mouvement, la vitesse angulaire sera nulle :

- $_{voile}$ = 0,0.

La vitesse horizontale sera donnée par le rapport entre la longueur de la course et le nombre d'images :

- $_{vieux} = ljk / n$.

Comme l'élément de liaison est horizontal, la vitesse verticale sera nulle :

- *velv* = 0,0.

Mouvement *F*

Soit $_{ni}$ le nœud de provenance du personnage, *nj* le nœud adjacent dans lequel le personnage va entrer, *RAIOF* le rayon de courbure du mouvement et *VELF* la vitesse maximale souhaitée (Figure 4).

Le personnage doit parcourir dans le sens rétrograde un arc de cercle de rayon *RAIOF* qui sous-tend un angle de θij radians. Les valeurs de l'angle implicite et de la longueur de l'arc peuvent être déterminées à l'aide du triangle rectangle représenté sur la figure.

La longueur de l'hypoténuse du triangle rectangle sera donnée par l'équation :

- *hanche* = rj - *bj* + *RAIOF*.

La longueur de l'obliquité transversale sera donnée par l'équation :

- *cattrans* = *wij* / 2.0 - $_{bij}$ + *RAIOF*.

L'angle sous-tendu et la longueur de l'arc à parcourir seront donnés par les équations :

- *ϑij* = arccos[7] *(cattrans* / *hip*) ;
- *cij* = *RAIOF* * θij.

Le nombre d'images qui composent l'animation et les valeurs des vitesses angulaire, horizontale et verticale seront donnés par les équations suivantes :

- *n* = ⌈cij / *VELF*⌉ ;
- $_{vela}$ = -θij / *n*[8] ;
- $_{old}$ = 2.0 * *RAIOF* * sin(θij / *n* / 2.0)[9] ;
- *velv* = 0,0.

Mouvement *B*

Soit *nj* le nœud duquel le personnage sortira, *nk* $_{le}$ nœud adjacent vers lequel le personnage se dirige, *RAIOB* le rayon de courbure du mouvement et *VELB* la vitesse maximale souhaitée (figure 5).

En appliquant le même raisonnement que pour la préparation du mouvement *F*, il s'avère que le personnage doit parcourir dans le sens rétrograde un arc de cercle de rayon *RAIOB*, ce qui implique une valeur de

---

[7] La fonction Math.acos() [4] peut être utilisée.
[8] Le signe négatif reflète la direction rétrograde prévue.
[9] Rappelez-vous que, dans chaque image, le personnage ne passera pas par un arc de cercle, mais par la chaîne correspondante.

angle de θjk radians. Les valeurs de l'angle implicite et de la longueur de l'arc peuvent être déterminées à l'aide du triangle rectangle représenté sur la figure.

La longueur de l'hypoténuse du triangle rectangle sera donnée par l'équation :

- *hip* = rj - *bj* + *RAIOB*.

La longueur de l'obliquité transversale sera donnée par l'équation :

- *cattrans* = *wjk* / 2,0 - *bjk* + *RAIOB*.

L'angle sous-tendu et la longueur de l'arc à parcourir seront donnés par les équations :

- *ϑjk* = arccos(*cattrans* / *hip*) ;
- *cjk* = *RAIOB* * θjk.

Le nombre d'images qui composent l'animation et les valeurs des vitesses angulaire, horizontale et verticale seront donnés par les équations suivantes :

- *n* = [cjk / *VELB]*;
- $_{voile}$ = -θjk / *n* ;
- $_{old}$ = 2.0 * *RAIOB* * sin(θjk / *n* / 2.0) ;
- *velv* = 0,0.

Mouvement *A*

Soit $_{ni}$ le nœud de provenance du personnage, *nj* le nœud adjacent où se trouve le personnage, *nk* le nœud adjacent où se dirige le personnage et *VELA* la vitesse maximale prévue pour ce mouvement (Figure 4 et Figure 5).

Le personnage doit parcourir dans la direction directe un arc de cercle de rayon (*rj* - *bj*) qui sous-tend un angle de ϕijk radians. L'angle sous-tendu sera obtenu en calculant la différence entre les orientations des arcs $_{ajk}$ et *aji* $_{e\,t\ \ e\,n}$ soustrayant les angles complémentaires[10] des angles θij et θjk :

- *φijk* = *αjk* - *αji* - (π / 2,0 - θij) - (π / 2,0 -

θjk). Sachant que αji = αij - π, on aura :

- *φijk* = αjk - (αij - π) - (π / 2,0 - θij) - (π / 2,0 - θjk) ;
  φijk = *αjk* - *αij* + π - π / 2,0 + θij - π / 2,0 + θjk
  ; ϕijk = αjk - αij + θij + θjk.

Compte tenu de la direction directe prévue du mouvement, il convient de s'assurer que ϕijk > 0,0, en ajoutant 2,0. * π si nécessaire.

En revanche, pour éviter que le caractère ne décrive des tours de cercle superflus, il convient de s'assurer que ϕijk $_{\leq}$ 2,0 * π, en soustrayant 2,0 * π si nécessaire.

[10] Deux angles sont dits complémentaires lorsque leur somme fait 90° ou π / 2 radians [3].

La longueur de l'arc sera donnée par l'équation :

- $_{dijk =}$ $(rj - bj) * \phi ijk$.

Le nombre d'images qui composent l'animation et les valeurs des vitesses angulaire, horizontale et verticale seront donnés par les équations suivantes :

- $n = \lceil dijk / VELA \rceil$;
- $_{vela} = \phi ijk / n$;
- $_{velh} = 2{,}0 * (rj - bj) * \sin(\phi ijk / n / 2{,}0)$;
- $velv = 0{,}0$.

Animation d'un mouvement élémentaire

L'orientation et l'emplacement du personnage dans l'image suivante seront donnés par les équations suivantes :

- $dir' = dir + {}_{sail}$;
- $x'_P = xP + {}_{ancien} * \cos(dir)$;
- $y'_P = yP + {}_{ancien} * \sin(dir)$;
- $z'_P = zP + velv$.

Étant donné la nature automatique du mouvement, la détection des collisions ne sera pas nécessaire, de sorte que l'orientation et la position du personnage peuvent être mises à jour sans condition :

- $dir = dir'$;
- $xp = x'_p$;
- $yp = y'_p$;
- $zp = z'_p$.

Le nombre d'images pour compléter l'animation doit alors être réduit d'une unité. Si ce numéro est annulé, l'animation en cours est terminée et l'animation du prochain mouvement élémentaire doit être préparée.

Initialisation de la position et de l'orientation du personnage

Pour déterminer la position et l'orientation du personnage (représenté en rouge sur la figure 6) au nœud d'origine nj, on peut imaginer qu'il a effectué un mouvement circulaire rétrograde en entrant dans le cercle du rond-point depuis un nœud adjacent $_{ni.}$

La valeur de l'angle impliqué par l'arc de cercle peut être déterminée à l'aide du triangle rectangle représenté sur la figure.

La longueur de l'hypoténuse du triangle rectangle sera donnée par l'équation :

- $hanche = \mathrm{rj} - bj + RAIOF$.

La longueur de l'obliquité transversale sera donnée par l'équation :

- $cattrans = wij / 2.0 - {}_{bij} + RAIOF$.

L'angle sous-tendu sera donné par l'équation :

- ϑij = arccos(*cattrans* / *hip*).

L'orientation sera obtenue en ajoutant à l'orientation de l'arc *aji* l'angle complémentaire de l'angle θij et également π / 2.0 radians :

- *dir* = *αji* + (π / 2 - θij) + π / 2.0 ;
  dir = *αji* - θij + π.

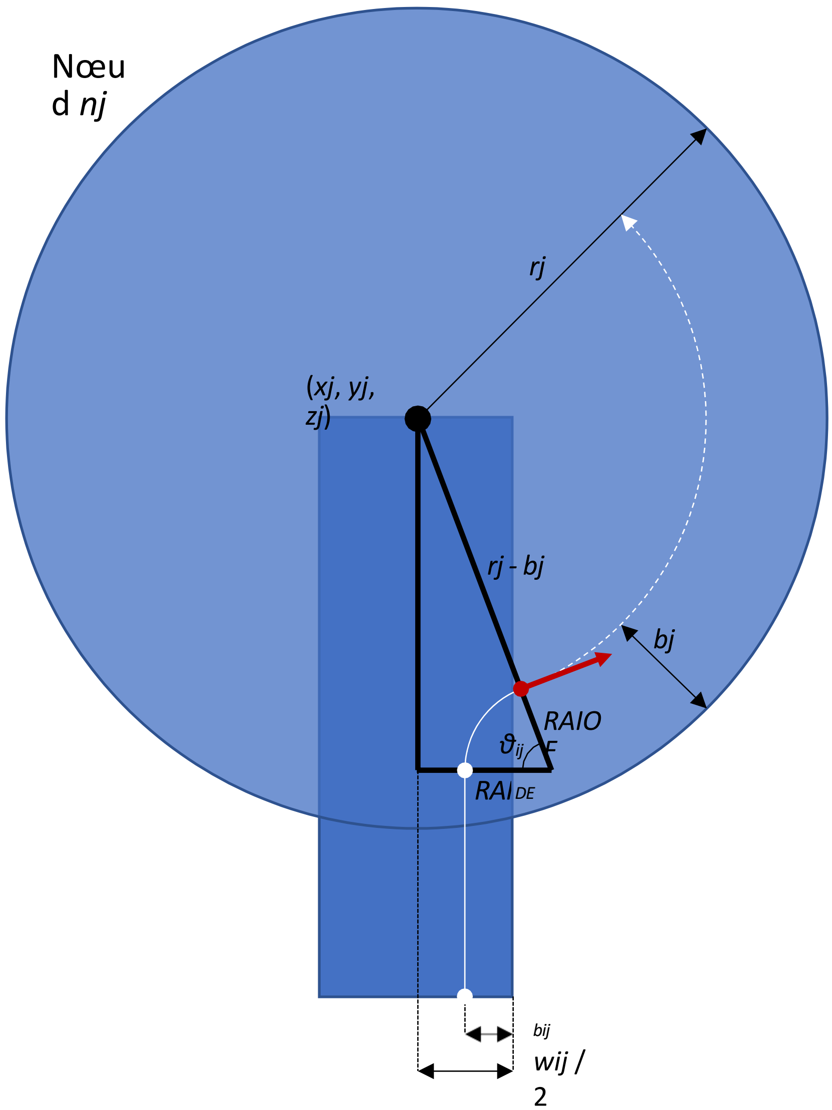

Figure 6 - Emplacement et orientation du caractère initial

Sachant que *αji* = *αij* - π, on aura :

- *dir* = *αij* - π - θij + π ;
  dir = *αij* - θij.

L'emplacement du personnage peut être calculé en utilisant le centre du cercle comme référence :

- $xP = xj + (rj - bj) * \sin(dir)$;
- $yP = yj - (rj - bj) * \cos(dir)$;
- $zP = zj + HEIGHT\_PERSONAGE / 2.0$.

Observations

Malgré le soin mentionné dans la section Mouvement *A* lors de la détermination de la valeur de ϕijk, il peut arriver que, dans le mouvement *A*, le personnage décrive un virage apparemment inutile autour du cercle d'un rond-point. Il ne s'agit pas d'une erreur. Il arrive que, selon les valeurs définies pour le rayon du cercle et les rayons de courbure des mouvements *F* et *B, il* soit physiquement impossible de parcourir la succession des mouvements *F*, *A* et *B* sans faire un tour supplémentaire. Ce comportement peut être évité en réduisant les valeurs de *RAIOF* et *RAIOB* et/ou en augmentant la valeur de *K_CIRCLE*. Si la première solution est choisie, les valeurs de *VELF* et *VELB* devront également être réduites, sinon l'animation des mouvements *F* et *B* perdra en fluidité.

Placement du personnage dans la scène

Selon la nature du personnage, il peut être nécessaire ou non de l'incliner sur les rampes constituant les arcs du graphique.

Si le personnage est bipède et se déplace à pied, sur un monocycle ou sur un *Segway* [2], par exemple, on peut supposer que sa posture ne s'écartera pas de manière significative de la verticale, et qu'il ne sera donc pas nécessaire de l'incliner. En revanche, les quadrupèdes, ou ceux qui se déplacent à vélo, en tricycle ou en voiture, entre autres exemples, devront être inclinés. Si *aij* est l'arc à parcourir, l'angle d'inclinaison sera le même que la rampe correspondante : βij.

Références

[1] Wikipedia, "Accord", [En ligne]. Disponible : https://en.wikipedia.org/wiki/Chord_(géométrie). [Accédé
le 20 septembre 2022].

[2] Wikipedia, "Segway", [En ligne]. Disponible : https://en.wikipedia.org/wiki/Segway. [consulté le 20 septembre 2022].

[3] Wikipedia, "Angle : Angles complémentaires", [En ligne]. Disponible : https://en.wikipedia.org/wiki/Angle#complementary_angle. [consulté le 20 septembre 2022].

[4] Mozilla, "Math.acos()", [En ligne]. Disponible : https://developer.mozilla.org/en-US/docs/Web/JavaScript/Référence/Global_Objects/Math/acos. [consulté le 20 septembre 2022].

[5] Mozilla, "Math.atan2()", [En ligne]. Disponible : https://developer.mozilla.org/en-US/docs/Web/JavaScript/Référence/Global_Objects/Math/atan2. [consulté le 20 septembre 2022].

[6] Mozilla, "Math.ceil()", [En ligne]. Disponible : https://developer.mozilla.org/en-US/docs/Web/JavaScript/Référence/Global_Objects/Math/ceil. [consulté le 20 septembre 2022].

[7] Three.js, "Object3D : up", [En ligne]. Disponible : https://threejs.org/docs/index.html?q=object3d#api/en/core/Object3D.up. [consulté le 20 septembre 2022].